

**GAME SIMULASI PEMBELAJARAN BUDIDAYA
TANAMAN JAGUNG BERBASIS *REAL TIME*
*STRATEGY***

Oleh

RIZKI MALAHEDI

T3116154

SKRIPSI

Untuk memenuhi salah satu syarat ujian
guna memperoleh gelar Sarjana



**PROGRAM SARJANA
TEKNIK INFORMATIKA
UNIVERSITAS ICHSAN GORONTALO
GORONTALO**

2020

PENGESAHAN SKRIPSI

**GAME SIMULASI PEMBELAJARAN BUDIDAYA
TANAMAN JAGUNG BERBASIS *REAL TIME*
*STRATEGY***

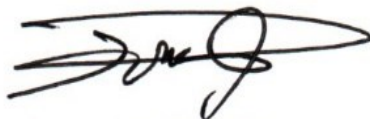
Oleh
RIZKI MALAHEDI
T3116154

SKRIPSI

Untuk memenuhi salah satu syarat ujian
Guna memperoleh gelar Sarjana
Program Studi Teknik Informatika,
Ini telah disetujui oleh Tim Pembimbing

Gorontalo, 1 Desember 2020

Pembimbing I



Irvan A Salihi, M.Kom
NIDN: 0928028101

Pembimbing II



Irma Surya Kumala Idris, M.Kom
NIDN: 0921128801

PERSETUJUAN SKRIPSI

GAME SIMULASI PEMBELAJARAN BUDIDAYA TANAMAN JAGUNG BERBASIS *REAL TIME* STRATEGY

Oleh

RIZKI MALAHEDI

T3116154

Diperiksa oleh Panitia Ujian Strata Satu (S1)

Universitas Ichsan Gorontalo

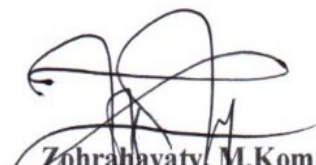
Gorontalo, 9 Desember 2020

1. Ketua Penguji
Budy Santoso, S.Kom, M.Eng
2. Anggota
Abd. Rahmat Karim Haba, M.Kom
3. Anggota
Warid Yunus, M.Kom
4. Anggota
Irvan Abraham Salihi, M.Kom
5. Anggota
Irma Surya Kumala Idris, M.Kom



Mengetahui :

Dekan Fakultas Ilmu Komputer



Zohrahayaty, M.Kom
NIDN.0912117702

Ketua Program Studi



Irvan Abraham Salihi, M.Kom
NIDN.0928028101

PERNYATAAN SKRIPSI

Dengan ini saya menyatakan bahwa:

1. Karya tulis (Skripsi) saya ini adalah asli dan belum pernah diajukan untuk mendapatkan gelar akademik (Sarjana) baik di Universitas Ichsan Gorontalo maupun di perguruan tinggi lainnya.
2. Karya tulis (Skripsi) saya ini adalah murni gagasan, rumusan, dan penelitian saya sendiri, tanpa bantuan pihak lain, kecuali arahan dari Tim Pembimbing.
3. Dalam Karya tulis (Skripsi) saya ini tidak terdapat karya atau pendapat yang telah dipublikasikan orang lain, kecuali secara tertulis dicantumkan pula dalam daftar pustaka.
4. Pernyataan ini saya buat dengan sesungguhnya dan apabila dikemudian hari terdapat penyimpangan dan ketidakbenaran dalam pernyataan ini, maka saya bersedia menerima sanksi akademik berupa pencabutan gelar yang telah diperoleh karena karya tulis ini, serta sanksi lainnya sesuai dengan norma-norma yang berlaku di Universitas Ichsan Gorontalo.

Gorontalo, 1 Desember 2020

Yang Membuat Pernyataan




Rizki Malahedi

ABSTRACT

Learning about plant cultivation is very important in the world of education, therefore in schools many are taught about plant cultivation with various kinds of learning methods, from manuals to direct field practice. However, in the process it all takes a long time and costs a lot. Therefore the authors took the initiative to create a learning simulation game application to be applied in the pursuit of plant cultivation, especially corn plants. The advantage of this game is that it is run offline and can make it easy for students to follow the lessons without requiring a long time and expensive costs in the catch-up process. The results showed that the researchers succeeded in making a game application that would facilitate students in learning the cultivation of corn plants.

Keywords: Simulation Games, Learning, Plant Cultivation, Corn

ABSTRAK

Pembelajaran terhadap budidaya tanaman sangat penting dalam dunia pendidikan, oleh karena itu di sekolah banyak diajarkan tentang budidaya tanaman dengan berbagai macam metode pembejaran, mulai dari buku panduan sampai praktik di lapangan secara langsung. Namun dalam proses itu semua membutuhkan waktu yang panjang dan biaya yang tidak sedikit. Oleh karena itu penulis berinisiatif membuat aplikasi game simulasi pembelajaran untuk diterapkan dalam pembejaran budidaya tanaman khususnya tanaman jagung. Kelebihan dari game ini adalah dijalankan dalam keadaan *offline* dan dapat membuat para siswa mudah untuk mengikuti pembelajaran tanpa harus membutuhkan waktu yang lama dan biaya yang mahal dalam proses pembejaran. Hasil penelitian menunjukkan bahwa peneliti berhasil membuat aplikasi game yang akan memudahkan siswa dalam pembelajaran budidaya tanaman jagung.

Kata kunci : Game Simulasi, Pembelajaran, Budidaya Tanaman, Jagung

KATA PENGANTAR

Alhamdulillah, penulis dapat menyelesaikan skripsi ini dengan judul “***Game Simulasi Pembelajaran Budidaya Tanaman Jagung Berbasis Real Time Strategi***” untuk memenuhi salah satu syarat Ujian Akhir guna memperoleh gelar Sarjana Komputer pada Program Studi Teknik Informatika Fakultas Ilmu Komputer Universitas Ichsan Gorontalo.

Penulis Menyadari sepenuhnya bahwa skripsi ini tidak mungkin terwujud tanpa bantuan dan dorongan dari berbagai pihak, baik bantuan moril maupun materil. Untuk itu, dengan segala keikhlasan dan kerendahan hati, penulis mengucapkan banyak terima kasih dan penghargaan yang setinggi-tingginya kepada:

1. Muhammad Ichsan Gaffar, S.E, M.Ak, selaku Ketua Yayasan Pengembangan Ilmu Pengetahuan dan Teknologi (YPIPT) Ichsan Gorontalo;
2. DR. Abdul Gaffar La Tjokke, M.Si, selaku Rektor Universitas Ichsan Gorontalo;
3. Zohrahayaty, S.Kom, M.Kom, selaku Dekan Fakultas Ilmu Komputer Universitas Ichsan Gorontalo;
4. Sudirman S Panna, S.Kom, M.Kom, selaku Wakil Dekan I Bidang Akademik Fakultas Ilmu Komputer Universitas Ichsan Gorontalo;
5. Irma Surya Kumala Idris, M.Kom, selaku Wakil Dekan II Bidang Administrasi Umum dan Keuangan Fakultas Ilmu Komputer Universitas Ichsan Gorontalo;
6. Sudirman Melangi, M.Kom, selaku Wakil Dekan III Bidang Kemahasiswaan Fakultas Ilmu Komputer Universitas Ichsan Gorontalo;
7. Irvan A Salihi, M.Kom, selaku Ketua Jurusan Teknik Informatika Fakultas Ilmu Komputer Universitas Ichsan Gorontalo;
8. Irvan A Salihi, M.Kom, selaku Pembimbing I;
9. Irma Surya Kumala Idris, M.Kom, selaku Pembimbing II;
10. Bapak dan Ibu Dosen Universitas Ichsan Gorontalo yang telah mendidik dan mengajarkan berbagai disiplin ilmu kepada penulis;

11. Kedua Orang Tua saya yang tercinta, atas segala kasih sayang, jerih payah dan doa restunya dalam membesarkan dan mendidik penulis;
12. Rekan-rekan seperjuangan yang telah banyak memberikan bantuan dan dukungan moril yang sangat besar kepada penulis;
13. Kepada semua pihak yang ikut membantu dalam penyelesaian proposal/skripsi ini yang tak sempat penulis sebutkan satu-persatu.

Semoga Allah, SWT melimpahkan balasan atas jasa-jasa mereka kepada kami. Penulis menyadari sepenuhnya bahwa apa yang telah dicapai ini masih jauh dari kesempurnaan dan masih banyak terdapat kekurangan. Oleh karena itu, penulis sangat mengharapkan adanya kritik dan saran yang konstruktif. Akhirnya penulis berharap semoga hasil yang telah dicapai ini dapat bermanfaat bagi kita semua, Aamiin.

Gorontalo, 1 Desember 2020

Penulis

DAFTAR ISI

PENGESAHAN SKRIPSI	ii
PERSETUJUAN SKRIPSI	iii
PERNYATAAN SKRIPSI.....	iv
<i>ABSTRACT</i>	v
ABSTRAK	vi
KATA PENGANTAR	vii
DAFTAR ISI.....	ix
DAFTAR GAMBAR	xii
DAFTAR TABEL.....	xiv
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Identifikasi Masalah.....	3
1.3 Rumusan Masalah	3
1.4 Tujuan Penelitian	3
1.5 Manfaat Penelitian	3
1.5.1 Manfaat Teoritis.....	3
1.5.2 Manfaat Praktis	3
BAB II LANDASAN TEORI.....	4
2.1 Tinjauan Studi.....	4
2.2 Tinjauan Pustaka	6
2.2.1 <i>Game</i>	6
2.2.2 <i>Game</i> Simulasi	8
2.2.3 Pembelajaran Berbasis Komputer	8
2.2.5 Budidaya Tanaman Jagung	9
2.2.6 <i>Unified Modeling Language (UML)</i>	12
2.2.7 Aplikasi Pendukung	17
2.2.8 Uji Kelayakan Portability	18
2.2.8 Pengujian Sistem.....	18
2.3 Kerangka Pikir	19

BAB III METODE PENELITIAN	20
3.1 Jenis, Metode, Subjek, Objek, Waktu, dan Lokasi Penelitian	20
3.2 Pengumpulan Data	20
3.2.1 Observasi.....	20
3.2.2 Wawancara.....	20
3.2.3 Studi Literatur	20
3.3 Pengembangan <i>Game</i>	21
3.3.1 Analisis Sistem.....	21
3.3.3 Pengujian Sistem.....	21
3.4 Implementasi <i>Game</i>	22
BAB IV HASIL PEMBAHASAN.....	23
4.1 Hasil Pengembangan <i>Game</i>	23
4.1.1 Rancangan Umum.....	23
4.1.2 Rancangan Level.....	23
4.1.3 Rancangan Game Budidaya Tanaman Jagung.....	23
4.1.4 <i>Storyboard</i>	23
4.1.5 Desain Karakter.....	28
4.1.6 Hasil Analisis Sistem	32
4.1.7 Hasil Uji Kelayakan Portability	42
4.1.8 Hasil Pengujian Sistem	43
BAB V PEMBAHASAN	48
5.1 Pembahasan Model	48
5.2 Pembahasan Sistem.....	48
5.2.1 Tampilan Menu Utama	48
5.2.2 Tampilan Mulai Permainan.....	49
5.2.3 Tampilan Kegiatan Mamacul Untuk Persiapan Penanaman.....	50
5.2.4 Tampilan Kegiatan Penanaman Tanaman Jagung	50
5.2.5 Tampilan Kegiatan Penyiraman Tanaman	50
5.2.6 Tampilan Kegiatan Pemupukan Tanaman	51
5.2.7 Tampilan Kegiatan Penyiangan Tanaman	52
5.2.8 Tampilan Kegiatan Pembasmian Hama.....	52

5.2.9 Tampilan Menu Petunjuk.....	52
5.2.10 Tampilan Menu Soal Latihan.....	53
5.2.11 Tampilan Menu Tentang Aplikasi	54
BAB VI PENUTUP	55
6.1 Kesimpulan	55
6.2 Saran	55
DAFTAR PUSTAKA	56
LAMPIRAN	

DAFTAR GAMBAR

Gambar 2.1: Kerangka Pikir	19
Gambar 4.1: Aktor Berjalan.....	28
Gambar 4.2: Aktor Mencangkul	28
Gambar 4.3: Aktor Menanam	28
Gambar 4.4: Aktor Menyiram.....	28
Gambar 4.5: Aktor Memupuk.....	29
Gambar 4.6: Aktor Menyemprot.....	29
Gambar 4.7: Hama Tanaman Jagung Belalang dan Ulat	29
Gambar 4.8: Cangkul	29
Gambar 4.9: Tanaman Jagung	30
Gambar 4.10: Air	30
Gambar 4.11: Pupuk Tanaman Jagung	30
Gambar 4.12: Obat Penyiangkan	31
Gambar 4.13: Obat Basmi Hama	31
Gambar 4.14: <i>Use Case Diagram Player</i>	32
Gambar 4.15: <i>Activity Diagram</i> Mulai Permainan	33
Gambar 4.16: <i>Activity Diagram</i> Kegiatan Memacul	33
Gambar 4.17: <i>Activity Diagram</i> Kegiatan Menanam.....	34
Gambar 4.18: <i>Activity Diagram</i> Kegiatan Menyiram	34
Gambar 4.19: <i>Activity Diagram</i> Kegiatan Memupuk	35
Gambar 4.20: <i>Activity Diagram</i> Kegiatan Penyiangkan	35
Gambar 4.21: <i>Activity Diagram</i> Kegiatan Basmi Hama.....	36
Gambar 4.22: <i>Activity Diagram</i> Petunjuk.....	36
Gambar 4.23: <i>Activity Diagram</i> Soal Latihan.....	37
Gambar 4.24: <i>Activity Diagram</i> Tentang Aplikasi	37
Gambar 4.25: Sequence Diagram Mulai Permainan.....	38
Gambar 4.26: <i>Sequence Diagram</i> Kegiatan Memacul	38
Gambar 4.27: <i>Sequence Diagram</i> Kegiatan Menanam.....	39
Gambar 4.28: <i>Sequence Diagram</i> Kegiatan Menyiram	39

Gambar 4.29: <i>Sequence Diagram</i> Kegiatan Memupuk	39
Gambar 4.30: <i>Sequence Diagram</i> Kegiatan Penyiangan	40
Gambar 4.31: <i>Sequence Diagram</i> Kegiatan Basmi Hama	40
Gambar 4.32: <i>Sequence Diagram</i> Petunjuk	40
Gambar 4.33: <i>Sequence Diagram</i> Soal Latihan	41
Gambar 4.34: <i>Sequence Diagram</i> Tentang Aplikasi	41
Gambar 4.35: Flowchart Pengujian White Box	46
Gambar 4.36: Flowgraph Pengujian White Box	47
Gambar 5.1: Tampilan Menu Utama	48
Gambar 5.2: Tampilan Mulai Permainan	49
Gambar 5.3: Tampilan Kegiatan Memacul	50
Gambar 5.4: Tampilan Kegiatan Penanaman Tanaman Jagung	50
Gambar 5.5: Tampilan Kegiatan Penyiraman Tanaman	51
Gambar 5.6: Tampilan Kegiatan Pemupukan Pertama	51
Gambar 5.7: Tampilan Kegiatan Penyiangan Pertama	52
Gambar 5.8: Tampilan Kegiatan Pengendalian Hama	52
Gambar 5.9: Tampilan Petunjuk Permainan	53
Gambar 5.10: Tampilan Soal Latihan	53
Gambar 5.11: Tampilan Tentang Aplikasi	54

DAFTAR TABEL

Tabel 2.1: Referensi Penelitian Terdahulu.....	4
Tabel 2.2: Simbol-simbol Use Case Diagram.....	12
Tabel 2.3: Simbol-simbol Activity Diagram	13
Tabel 2.4: Simbol-simbol Sequence Diagram	14
Tabel 2.5: Multiplicity Class Diagram.....	15
Tabel 2.6: Simbol-simbol Class Diagram.....	16
Tabel 4.1: Storyboard bagian Menu Utama	23
Tabel 4.2: Storyboard bagian Mulai Permainan	24
Tabel 4.3: Storyboard Sublevel.....	27
Tabel 4.4: Hasil Pengujian Black Box	43

BAB I

PENDAHULUAN

1.1 Latar Belakang

Di salah satu provinsi yang ada di Indonesia, sekelompok mahasiswa melalui kegiatan Kuliah Kerja Nyata Pembelajaran Pemberdayaan Masyarakat (KKN PPM) mengaplikasikan dan mengembangkan ilmu pengetahuan dan teknologi yang dikuasai di tengah masyarakat khususnya pada anak usia sekolah, yaitu siswa dibimbing dan diperkenalkan pada konsep berwirausaha budidaya tanaman pangan, salah satunya budidaya tanaman jagung. Kegiatan ini bertujuan agar pembudidayaan tanaman jagung itu dapat dikembangkan sendiri oleh para siswa secara langsung[1].

Kegiatan ini dilakukan dengan cara mendampingi siswa dalam rangkaian proses budidaya tanaman secara langsung di lapangan dengan memanfaatkan halaman sekolah dengan menggunakan alat dan bahan yang telah disediakan sebelumnya. Alat dan bahan yang dibutuhkan antara lain cangkul untuk persiapan pembuatan bedengan, benih tanaman, serta pupuk yang akan digunakan dalam perawatan tanaman. Dalam kegiatan ini seluruh mahasiswa KKN PPM berperan aktif dalam mengajarkan siswa secara langsung mulai dari proses pembenihan, persiapan bedengan, pemberian pupuk, pemeliharaan serta penanganan pascapanen[1]. Pelaksanaan kegiatan ini membutuhkan waktu yang cukup panjang, karena lamanya waktu budidaya tanaman jagung sekitar 80 - 150 hari baru bisa dilakukan pemanenan[2].

Kegiatan ini sangat menarik, karena para siswa dapat mempraktekkan langsung apa yg telah diajarkan, namun berdasarkan itu pula masih terdapat beberapa kendala yang muncul. Kendala yang dihadapi dalam pelatihan ini adalah masih mahal nya alat dan bahan yang harus disiapkan dalam proses pembudidayaan, banyaknya elemen mahasiswa dan guru yang harus berperan aktif dalam proses pembelajaran dan pendampingan secara langsung, serta waktu yang diperlukan dalam proses pembudidayaan cukup lama.

Dari masalah yang muncul diatas, maka cara yang akan digunakan untuk mempercepat proses pembelajaran budidaya tanaman jagung adalah dengan cara menuangkan proses pembelajaran budidaya tersebut dalam sebuah *game*. *Game* saat ini bukan hanya memberikan hiburan semata namun juga bisa menjadi sarana pembelajaran yang sangat menarik[3]. Salah satunya adalah game simulasi. Jenis game ini bertujuan mensimulasikan kehidupan secara nyata dan detail, dimana menekankan kebebasan pemain untuk melakukan sesuatu seperti pada kehidupan sehari-hari[4].

Dalam penelitian ini, peneliti akan mencoba merancang sebuah game simulasi dengan memperhatikan parameter-parameter yang sekiranya sangat berpengaruh dalam proses pembelajaran budidaya tanaman jagung itu sendiri. Game akan dimainkan dan diperankan langsung oleh para siswa mulai dari penyiapan alat dan bahan yang diperlukan, proses penyiapan lahan, penanaman benih, perawatan tanaman, serta pemanenan seperti yang terjadi pada kehidupan secara nyata. Oleh karena itu, dengan adanya game ini siswa tidak lagi harus membeli alat dan bahan secara langsung, dimana siswa tidak lagi harus mengeluarkan biaya yang mahal dalam proses pembelajaran budidaya tanaman jagung. Para pengajarpun tidak lagi harus turun lapangan secara langsung dalam mengajarkan dan membimbing siswa dalam pembelajaran budidaya tanaman jagung itu sendiri. Dalam proses pembelajaran melalui game simulasi ini juga tidak membutuhkan waktu yang panjang, siswa cukup hanya memainkan game dengan mengikuti langkah demi langkah yang ada pada game tersebut.

Berdasarkan masalah dan pengertian dari game simulasi tersebut, oleh karena itu peneliti tertarik untuk melakukan penelitian dengan judul **“*Game Simulasi Pembelajaran Budidaya Tanaman Jagung Berbasis Real Time Strategi*”**. Diharapkan penelitian ini dapat memberi banyak pengetahuan tentang pembudidayaan tanaman jagung serta menambah minat para siswa dalam proses pembelajaran.

1.2 Identifikasi Masalah

Proses pembelajaran budidaya tanaman jagung yang digunakan masih menggunakan metode pengajaran dan pendampingan praktek dilapangan secara langsung, sehingga masih harus memerlukan alat dan bahan serta pengajar dan pembimbing yang tidak sedikit, ditambah lagi dengan membutuhkan waktu yang panjang.

1.3 Rumusan Masalah

Bagaimana *game* simulasi pembejaran budidaya tanaman jagung berbasis *real time strategy* dapat memberikan pembelajaran dan pengetahuan terhadap budidaya tanaman jagung dengan seefisien mungkin, baik dari segi biaya yang murah dan waktu yang singkat .?

1.4 Tujuan Penelitian

Game simulasi pembejaran budidaya tanaman jagung berbasis *real time strategy* dapat memberikan pembelajaran dan pengetahuan terhadap budidaya tanaman jagung dengan seefisien mungkin, baik dari segi biaya yang murah dan waktu yang singkat.

1.5 Manfaat Penelitian

1.5.1 Manfaat Teoritis

Memberikan masukan bagi perkembangan ilmu pengetahuan dan teknologi, khususnya pada bidang ilmu komputer.

1.5.2 Manfaat Praktis

Sumbangan pemikiran, karya, bahan pertimbangan, atau solusi bagi *game developer* guna mendukung pengambilan keputusan dalam rangka menghasilkan *game* yang berkualitas.

BAB II

LANDASAN TEORI

2.1 Tinjauan Studi

Adapun penelitian terdahulu yang berhubungan dengan topik peneliti untuk dijadikan sebagai bahan referensi nantinya.

Tabel 2.1: Referensi Penelitian Terdahulu

NO	PENELITI	JUDUL	TAHUN	METODE	HASIL
1.	Haryanto, Kristo Radion Purba, Kartika Gunadi	Pembuatan Strategy Farming Game Berbasis Flash	2016		Dari hasil kuesioner dengan rata – rata 7.4, membuktikan game dengan tipe strategi dan bercocok tanam jika digabungkan akan menjadi game yang cukup menarik
2.	Christian Danny Irawan, Dringhuzen J. Mamahit, Alwin M. Sambul	Pembuatan Game Simulasi Kewirausa haan untuk Profesi Petani	2019		Hasil kuesioner yang didapatkan dari 30 responden secara online tentang pendapat terhadap game simulasi kewirausahaan yang telah dibuat, responden menjawab sebanyak 83,4% untuk menyimulasikan kewirausahaan dengan

					baik, untuk penyampaian informasi kewirausahaan sebanyak 83,3%, alur cerita yang menarik sebanyak 73,3% dan untuk minigame yang ditawarkan sebanyak 90%. Kemudian hasil kuesioner yang dimana menguji apakah game simulasi kewirausahaan ini dapat memberikan manfaat kepada pengguna dalam mempelajari kewirausahaan terjadi peningkatan pada jumlah jawaban yang benar dari sebelum hingga setelah pengguna memainkan game simulasi kewirausahaan, yang dimana pada soal nomor 1 sebanyak 40%, soal nomor 2 tidak terjadi peningkatan, soal
--	--	--	--	--	--

					nomor 3 sebanyak 30%, soal nomor 4 sebanyak 20% dan soal nomor 5 sebanyak 30%
3.	Verra E. J. Mawitjere, Revolson A. Mege, Julduz R. Paus, Jeane Mantiri, Mokosuli Y. Samuel	Menanamkan Jiwa Kewirausahaan melalui Percontohan Budidaya Tanaman Pangan pada Anak Usia Sekolah	2018		Hasil yang dicapai dalam kegiatan KKN PPM ini yaitu, terfasilitasi percontohan budidaya tanaman pangan di samping sebagai upaya menunjang pemenuhan kebutuhan konsumsi juga sekaligus telah menjadi media pembelajaran pendidikan ekonomi dalam upaya menanamkan jiwa kewirausahaan, juga menjadi sumber biaya kebutuhan sekolah bagi anak usia sekolah

2.2 Tinjauan Pustaka

2.2.1 *Game*

Game merupakan kata yang berasal dari bahasa Inggris yang memiliki arti permainan. Permainan merupakan sesuatu yang dapat dimainkan yang dimana pada permainan tersebut memiliki aturan-aturan tertentu sehingga ada yang

menang dan ada yang kalah[5]. Game dibuat dengan teknik dan metode tertentu yang dapat memberi kesenangan dan kepuasan batin bagi para pemainnya. Game mempunyai potensi yang bisa dikatakan sangat besar untuk membangun sebuah motivasi dalam proses pembelajaran[6]. Berdasarkan genre-nya, game dibagi menjadi beberapa jenis, seperti[7]:

- a) *Action* : Sebuah permainan yang didalamnya terdapat sebuah rintangan dan membutuhkan pemain yang memiliki teknik yang baik dalam menyelesaikan berbagai rintangan tersebut dengan kecepatan reflex, akurasi, dan ketepatan waktu
- b) *Fighting* : Sebuah permainan pertarungan antara dua karakter yang dimana karakter lawan dikendalikan oleh komputer.
- c) *Shooter* : Jenis permainan yang merupakan turunan dari permainan aksi, namun banyak para pemain yang beranggapan bahwa jenis game ini merupakan jenis game yang berdiri sendiri.
- d) *Racing* : Permainan yang menampilkan dimana pemain dituntut untuk menunjukkan keterampilan dalam mengemudi pada suatu kompetisi balapan. Biasanya menggunakan motor atau mobil.
- e) *Sport* : Permainan olahraga yang didalamnya menuntut keterampilan dari pemain pada sebuah pertandingan olahraga tersebut, seperti basket, tenis, sepak bola, dan lainnya.
- f) *Adventure* : Permainan yang menampilkan sebuah rintangan berjangka panjang yang menuntut pemain mengatasi berbagai rintangan yang ada dengan menggunakan alat atau item sebagai alat bantu dalam mengatasi rintangan tersebut. *Game* ini menggabungkan unsur-unsur jenis komponen antara game *action* dan game *adventure*.
- g) *Strategi* : Jenis permainan yang hampir sejenis dengan game simulasi yang dimana game jenis ini memiliki tujuan yang jelas, sehingga menekankan pemain dalam penggunaan strategi juga melibatkan masalah strategi, taktik, dan logika.

- h) *Role Playing Game* : Jenis permainan yang dimana para pemainnya memainkan peran tokoh berdasarkan hasil imajinasi dan berkolaborasi dalam merajut sebuah cerita bersama.
- i) *Raal Time Strategi* : Jenis ini adalah permainan dimana para pemain dihadapkan untuk melakukan sesuatu gerakan dan yang tercepatlah yang akan menang[6].

2.2.2 *Game Simulasi*

Game simulasi adalah *game* yang didalamnya menerapkan keadaan-keadaan berdasarkan kehidupan sebenarnya[8]. *Game* simulasi memiliki penggambaran konsep permainan akan segala suatu hal pada kehidupan nyata, sehingga hal yang terjadi pada kehidupan sehari-hari akan lebih banyak ditemui di dalam *game* tipe simulasi. *Game* dengan tipe simulasi dapat menggambarkan tentang kehidupan pribadi sehari-hari, kehidupan pekerjaan serta pengaturan dan pengoperasian alat-tertentu[9].

2.2.3 *Pembelajaran Berbasis Komputer*

Pembelajaran berbasis komputer adalah program pembelajaran dimana dalam proses pembelajarannya menggunakan software komputer dalam bentuk program komputer yang didalamnya berisikan tentang muatan pembelajaran yang meliputi: judul pembelajaran, tujuan dari pembelajaran, materi pembelajaran, dan evaluasi pembelajaran.

Dalam proses pembelajaran yang dengan pembelajaran tersebut menggunakan media komputer, dapat tercipta suatu kemungkinan proses pembelajaran yang berlangsung secara individual. Dalam proses ini juga dapat menciptakan tumbuhnya pembelajaran secara mandiri dikalangan para siswa, sehingga para siswa tersebut akan mengalami proses yang lebih bermakna dibanding dengan pembelajaran konvensional. Adapun manfaat komputer untuk tujuan pendidikan diantaranya yaitu[10]:

- a) Komputer dapat berinteraksi dengan siswa secara perorangan seperti dalam hal bertanya dan menilai jawaban. Dengan kata lain tingkat

kecepatan belajar siswa dapat disesuaikan dengan tingkat penguasaannya, sebab kendali sepenuhnya berada ditangan siswa.

- b) Komputer dapat memberikan pengertian dan pemahaman terhadap siswa yang lamban menerima pelajaran, sebab komputer dapat menghadirkan suatu keadaan ataupun situasi yang lebih bersifat afektif secara individual, tidak adanya kebosanan, tidak pernah lupa, dan sangat sabar dalam menjalankan suatu perintah seperti yang diinginkan program yang dijalankan.
- c) Perkembangan setiap siswa selalu dapat dipantau serta kemampuan merekam siswa selama menggunakan suatu program pembelajaran memberikan kesempatan kepada siswa untuk jauh lebih baik dalam pembelajaran secara perorangan.
- d) Komputer bisa memberikan suatu pengarahan kepada siswa dalam hal mengerjakan latihan, melakukan kegiatan simulasi sebab dalam komputer menyediakan animasi grafik, warna dan musik yang dapat menambah realisme atau suatu keadaan secara nyata.

2.2.5 Budidaya Tanaman Jagung

Jagung merupakan salah satu tanaman pangan dunia yang terpenting, selain gandum dan padi[2]. Jagung adalah salah satu biji-bijian yang mempunyai kedudukan penting di sektor perekonomian dunia di kurun waktu abad 20 dan 21. Negara-negara produsen jagung yang utama di dunia saat ini adalah Amerika, China, Argentina, Meksiko, dan termasuk juga Indonesia. Ada beberapa daerah yang ada di Indonesia sebagai penghasil jagung antara lain adalah Sumatera Utara, Riau, Sumatera Selatan, Lampung, Jawa Barat, Jawa Tengah, D.I. Yogyakarta, Jawa Timur, Nusa Tenggara Timur, Sulawesi Utara, Sulawesi Selatan, dan Maluku[11]. Tanaman jagung merupakan salah satu tanaman yang memiliki banyak fungsi diantaranya yaitu sebagai sumber penghasil karbohidrat, pakan ternak baik itu batang dan daun yang masih hijau, biji yang dapat diambil minyaknya dan dijadikan tepung yang dikenal dengan tepung jagung atau maizena[2].

Teknik Budidaya

a) Penyiapan Lahan

Penyiapan lahan dilakukan dengan proses pengolahan tanah terlebih dahulu. Dalam proses pengolahan tanah ini dapat dilakukan dengan olah tanah sempurna dan jika tanahnya gembur tidak lagi dilakukan pengolahan tanah[12].

b) Penanaman

Pada proses penanaman tanaman jagung, jarak tanam yang dianjurkan ada 2 cara yaitu 70 cm x 20 cm dengan 1 benih/lubang tanam atau 75 cm x 40 cm dengan 2 benih/lubang tanam. Pada penanaman tanpa olah tanah bisa dilakukan langsung dicangkul atau koak tempat menugal benih, lalu beri pupuk kandang atau kompos 1-2 genggam atau selebihnya 50 gr tiap cangkulan atau koakan. Jika penanaman dilakukan dengan olah tanah sempurna cukup ditugal untuk dibuat lubang tanam benih, selanjutnya diberikan pupuk kandang atau kompos 1-2 genggam atau selebihnya 50 gr. Pemberian pupuk kandang ini dilakukan pada saat proses penanaman sebagai penutup benih yang baru ditanam. Bisa juga 3-7 hari sebelum dilakukan penanaman[12].

c) Pemupukan

Pemupukan dasar dapat dilakukan bersamaan dengan penanaman atau 7-10 hari setelah penanaman, yaitu seluruh pupuk SP-36, KCI dan 1/2 bagian Urea dengan cara ditugal 5 cm dari lubang tanaman. Kemudian pada pemupukan susulan pada saat umur tanaman 1 bulan setelah penanaman, yaitu 1/2 bagian Urea dengan cara tugal sedalam 5-10 cm ditutup kembali[2].

d) Penyiangan

Selama masa pertumbuhan tanaman jagung ini, dilakukan 2 kali proses penyiangan, yaitu saat tanaman berumur 2 minggu dan 4 minggu setelah tanam[2].

e) Pengendalian Hama dan Penyakit

Penyakit pada tanaman jagung yang sering dijumpai adalah penyakit yang dimana daunnya berwarna kuning atau penyakit bulai. Pengendalian penyakit bulai dengan perlakuan benih, 1 kg benih dicampur dengan metalaksis (Ridhomil atau Saromil) 2 gr yang dilarutkan dalam 7,5-10 ml air. Penyakit bulai ini juga disebabkan oleh jamur (*Fusarium sp*) yang menyerang saat tanaman jagung mulai tumbuh. Pengendalian untuk jamur dapat ini dilakukan sesaat setelah muncul gejala infeksi jamur, yaitu dilakukan penyemprotan dengan Fungisida (Dithane M-45) dengan dosis 45 gr / tank isi 15 liter. Dalam proses penyemprotan ini harus dilakukan penyemprotan pada bagian bawah tanaman atau di bawah tongkol tanaman jagung. Hama yang umum mengganggu pertanaman jagung berikutnya adalah dari jenis serangga, diantaranya adalah lalat bibit dan ulat penggerek batang dan tongkol. Pengendalian hama lalat bibit ini harus dilakukan pada proses penanaman, karena hama jenis ini umumnya mengganggu disaat awal pertumbuhan tanaman jagung itu sendiri. Pengendaliannya dapat menggunakan insektisida carbofuran terutama pada daerah yang diman disitu memang sering terjadi serangan lalat bibit. Untuk hama ulat penggerek batang, pengendaliannya dapat dilakukan dengan memberikan insektisida carbofuran sebanyak 3-4 butir dengan cara ditugal bersamaan saat pemupukan atau juga dapat dilakukan dengan disemprot insektisida cair fastac atau regent sesuai dengan dosis yang tertulis pada kemasan. Namun jika mulai muncul ada gejala serangan dapat dilakukan dengan pemberian carbofuran 3-4 butir/tanaman melalui pucuk tanaman yang mulai terserang[12].

f) Pemanenan

Pemanenan dapat dilakukan saat tanaman jagung berumur 80 – 150 hari setelah tanam yang ditandai dengan daun jagung atau klobot berwarna keputihan atau coklat dan mengering dengan biji mengkilap dan kadar air 25-30 %[2].

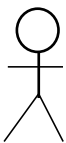
2.2.6 Unified Modeling Language (UML)

Unified Modeling Language atau UML adalah standarisasi bahasa pemodelan yang dibangun dengan menggunakan teknik pemrograman berorientasi objek dalam membangun sebuah perangkat lunak[13]. UML juga merupakan alat untuk mendukung pengembangan sistem[14]. Alat bantu yang digunakan dalam perancangan berorientasi objek berbasis UML adalah sebagai berikut:

a) *Use Case Diagram*

Use case diagram merupakan gambaran yang digunakan untuk merujuk pada kelakuan sistem[15]. Simbol-simbol yang digunakan dalam Use Case Diagram yaitu:

Tabel 2.2: Simbol-simbol *Use Case Diagram*[14]

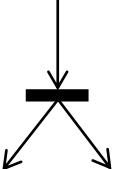
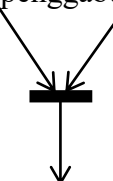
Gambar	Keterangan
<i>Use case</i> 	Menggambarkan fungsionalitas yang disediakan oleh sistem sebagai unit yang berfungsi sebagai penukar pesan antar unit dengan aktor.
Aktor/<i>Actor</i> 	Orang atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi. aktor berinteraksi dengan Use Case, tetapi tidak memiliki kontrol terhadap use case.
	Asosiasi antara aktor dan use case yang digambarkan dengan garis tanpa panah memberikan petunjuk atau isyarat akan siapa atau apa yang meminta interaksi secara langsung dengan sistem dan bukannya memberikan isyarat akan data.
	Asosiasi antara aktor dan use case yang menggunakan panah terbuka untuk mengisyaratkan bahwa terjadinya interaksi aktor dengan sistem secara pasif.

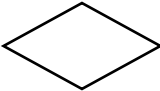
----- <<include>>	Pemanggilan use case oleh use case lain. Seperti pada halnya pemanggilan sebuah fungsi program.
<----- <<extends>>	Perluasan dari use case lain apabila suatu kondisi atau syarat telah terpenuhi.

b) *Activity Diagram* (Diagram Aktivitas)

Activity Diagram menggambarkan aktivitas dari sebuah sistem atau proses menu yang terdapat pada aplikasi pada perangkat lunak yang dibuat[14]. *Activity diagram* menggambarkan berbagai alir aktivitas dalam sebuah sistem dari satu aktivitas ke aktivitas lainnya[16]. Dalam hal ini simbol-simbol yang digunakan yaitu:

Tabel 2.3: Simbol-simbol *Activity Diagram*[14]

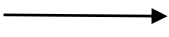
Gambar	Keterangan
<i>Start Point</i> 	Awal aktivitas pada sebuah diagram aktivitas. Biasanya diletakkan pada pojok kiri atas.
<i>End Point</i> 	Status akhir aktivitas dari sebuah sistem.
<i>Activities</i> 	Proses atau kegiatan yang dilakukan sistem
<i>Fork/percabangan</i> 	Menunjukkan suatu kegiatan yang dilakukan secara paralel atau untuk menggabungkan dua kegiatan paralel menjadi satu
<i>Join/penggabungan</i> 	Digunakan untuk menunjukkan saat terjadinya suatu perubahan pada sistem

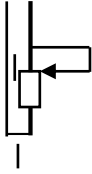
<i>Decision Points</i> 	Menggambarkan terjadinya kondisi dimana adanya pilihan tru atau false sehingga harus dilakukan pilihan untuk pengambilan keputusan
<i>Swimlane</i> 	Menggambarkan untuk menunjukkan siapa melakukan apa dalam proses pembagian aktivitas diagram

c) *Sequence Diagram* (Diagram Urutan)

Sequence diagram adalah suatu diagram yang menampilkan interaksi antar objek dengan objek lainnya yang terdapat dalam sistem yang disusun pada sebuah urutan atau rangkaian waktu[15]. Dalam hal ini simbol-simbol yang digunakan dalam *Sequence Diagram* yaitu:

Tabel 2.4: Simbol-simbol *Sequence Diagram*[14]

Gambar	Keterangan
<i>Entity Class</i> 	Gambaran awal sistem dan menjadi landasan untuk menyusun basis data yang terbentuk dari kumpulan kelas berupa entitas dari bagian sistem.
<i>Boundary Class</i> 	Sekumpulan kelas yang didalamnya terjadi suatu interaksi antara aktor dengan sistem, baik satu aktor atau lebih
<i>Control class</i> 	Berisikan logika aplikasi yang berasal dari suatu objek yang dimana logika aplikasi tersebut tidak memiliki tanggung jawab kepada entitas
<i>Message</i> 	Digunakan untuk proses mengirimkan pesan antar <i>class</i> yang terdapat pada sistem

<i>Recursive</i> 	Menggambarkan proses dimana terjadinya pengiriman pesan yang dikirim namun hanya tertuju pada dirinya sendiri
<i>Activation</i> 	Mewakili sebuah eksekusi operasi dari objek, durasi aktivasi sebuah operasi harus berbanding lurus dengan panjang kotak.
<i>Lifeline</i> 	Garis titik-titik yang merupakan garis yang terhubung dengan objek, disepanjang <i>lifeline</i> terdapat sebuah <i>activation</i>

d) *Class Diagram* (Diagram Kelas)

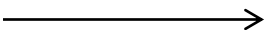
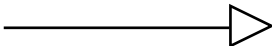
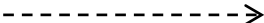
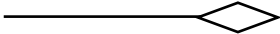
Class Diagram merupakan hubungan antar kelas serta penjelasan dari setiap kelas yang ada pada model desain dari suatu sistem. Hubungan antar kelas mempunyai keterangan yang dikenal dengan sebutan *Multiplicity* atau *Cardinality*.

Tabel 2.5: *Multiplicity Class Diagram*[14]

<i>Multiplicity</i>	Keterangan
1	Memiliki nilai yang tetap yaitu hanya satu
0..*	Boleh tidak ada atau 1 ataupun lebih dari satu
1..*	1 atau lebih dari satu
0..1	Boleh tidak ada, namun dibatasi maksimal 1
n..n	Batasan antara. Contohnya 6..8 memiliki artian minimal 2 maksimal 4

Dalam hal ini simbol-simbol yang digunakan dalam *Class Diagram* yaitu:

Tabel 2.6: Simbol-simbol *Class Diagram*[13]

Gambar	Keterangan
Kelas Nama_kelas +atribut +operasi()	Kelas yang akan digunakan dalam struktur yang terdapat pada sistem.
Antarmuka/<i>interface</i>  Nama_interface	Sama dengan konsep interface dalam pemrograman berorientasi objek.
Asosiasi/<i>association</i> 	Biasanya disertai dengan <i>multiplicity</i> yang merupakan suatu relasi antarkelas dalam artian umum.
Asosiasi berarah/ <i>directed association</i> 	Biasanya juga disertai dengan <i>multiplicity</i> yang merupakan relasi antarkelas dalam artian kelas yang satu digunakan oleh kelas yang lain.
Generalisasi 	Relasi antarkelas dalam artian generalisasipesialisasi (umum khusus).
Kebergantungan/ <i>Dependency</i> 	Relasi antarkelas dalam artian adanya kebergantungan antarkelas.
Agregasi/<i>aggregation</i> 	Relasi antarkelas dalam artian seluruh-bagian.

2.2.7 Aplikasi Pendukung

a) *Adobe Photoshop*

Adobe Photoshop merupakan perangkat lunak komputer yang bergerak pada program desain grafis yang banyak digunakan untuk keperluan desain dan fotografi, terutama dalam hal pengolahan objek yang berbasis gambar[17]. Aplikasi *Adobe Photoshop* juga di gunakan untuk memanipulasi foto sehingga hasil foto lebih menarik[18].

Adapun kelebihan dari *Adobe Photoshop* ini diantaranya dapat menghadirkan tekstur dan material yang beragam, dan dapat digunakan untuk mengedit foto dan gambar yang sudah ada sebelumnya. Sedangkan kelemahan dari *Adobe Photoshop* yaitu hanya dapat digunakan untuk menciptakan *Image* yang statis, dan juga pada perkembangan *Photoshop* itu sendiri yang selalu *update* sehingga spesifikasi Komputer untuk menjalankan program *Adobe Photoshop* tersebut juga harus sudah tinggi dan yang pastinya akan diimbangi oleh harga yang tinggi pula[18].

b) *Python*

Python adalah salah satu bahasa pemrograman tingkat tinggi atau *High Level Language*[19]. *Python* merupakan bahasa pemrograman yang bisa dikatakan mudah untuk dipelajari. *Python* dapat dijalankan di berbagai sistem operasi, seperti Linux, Mac OS, Windows, dan lain lain[20].

Adapun kelebihan dari bahasa pemrograman *Python* diantaranya yaitu: *Python* merupakan salah satu bahasa pemrograman yang mudah untuk dipelajari dan cepat dalam membantu programmer pada proses pembuatan aplikasi, *Coding* yang dibuat dengan menggunakan bahasa pemrograman *Python* mudah dibaca oleh programmer. Sedangkan kekurangan dari bahasa pemrograman *Python* yaitu terdapat pada kecepatan saat menjalankan aplikasi dimana kecepatannya tidak secepat

aplikasi yang dibuat dengan menggunakan bahasa pemrograman C atau C++[19].

2.2.8 Uji Kelayakan Portability

Portability atau Portabilitas merupakan salah satu uji kelayakan pada perangkat lunak. Portabilitas berasal dari kata port yang berarti sesuatu yang menjadi sarana komputer untuk berkomunikasi dengan dunia luar. Portabilitas merupakan kemudahan suatu untuk dipindahkan atau dioperasikan dari suatu lingkungan ke lingkungan yang lain[21].

2.2.8 Pengujian Sistem

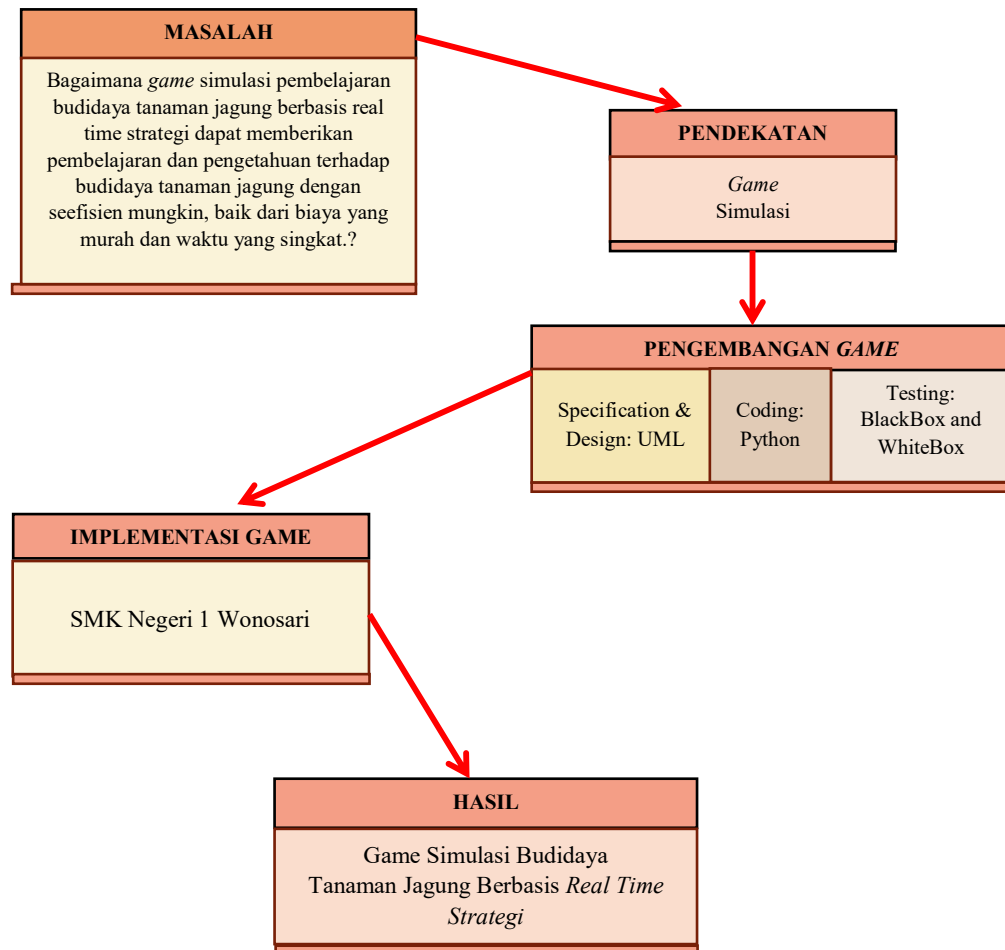
a) *Black-Box*

Black-Box Testing merupakan metode pengujian yang di mana pengujiannya berasal dari persyaratan fungsional atau dengan kata lain pengujian ini lebih mengutamakan fungsi dan kebergunaan yang ditentukan tanpa memperhatikan struktur program akhir[22].

b) *White-Box*

White-Box Testing merupakan metode pengujian yang dimana proses pengujiannya dengan cara melihat modul untuk dapat meneliti dan menganalisa kode dari program yang di buat apakah ada kesalahan atau tidak[23].

2.3 Kerangka Pikir



Gambar 2.1: Kerangka Pikir

BAB III

METODE PENELITIAN

3.1 Jenis, Metode, Subjek, Objek, Waktu, dan Lokasi Penelitian

Dipandang dari tingkat penerapannya, maka penelitian ini merupakan penelitian terapan.

Penelitian ini menggunakan metode penelitian eksperimen. Dengan demikian jenis penelitian ini adalah penelitian eksperimental.

Subjek penelitian ini adalah optimalisasi pada objek simulasi budidaya tanaman jagung berbasis *real time strategi*. Penelitian ini dimulai dari proses penyiapan alat dan bahan yang diperlukan, proses penyiapan lahan, penanaman benih, perawatan tanaman, serta pemanenan seperti halnya yang terjadi pada kehidupan secara nyata yang akan diimplementasikan pada proses pembelajaran budidaya tanaman jagung di SMK Negeri 1 Wonosari.

3.2 Pengumpulan Data

3.2.1 Observasi

Pengumpulan data dengan melakukan aktivitas pengamatan terhadap suatu objek secara cermat dengan mencatat secara sistematis tentang objek yang diteliti dengan maksud untuk mendapatkan informasi yang diperlukan dan dilakukan secara langsung di lokasi penelitian. Dalam hal ini, peneliti melakukan observasi di sekolah SMK Negeri 1 Wonosari dengan melakukan wawancara langsung pada guru pengajar maupun siswa jurusan pertanian.

3.2.2 Wawancara

Pengumpulan data dengan melakukan tanya jawab secara langsung pada narasumber yang ada, yaitu wawancara dengan guru maupun siswa jurusan pertanian yang nantinya akan menggunakan *game* simulasi ini.

3.2.3 Studi Literatur

Pengumpulan data dengan melakukan pencarian terhadap berbagai sumber tertulis diantaranya adalah buku-buku dan jurnal yang relevan dengan topik

maupun permasalahan yang dikaji, yaitu tentang *game* dan permasalahan tentang pembelajaran budidaya tanaman pangan terutama tanaman jagung.

3.3 Pengembangan *Game*

3.3.1 Analisis Sistem

Aanalisi sistem menggunakan pendekatan berorientasi objek menggunakan alat bantu UML yang digambarkan dalam bentuk:

- a) *Use Case Diagram*
- b) *Actifity Diagram*
- c) *Sequence Diagram*

3.3.2 Uji Kelayakan Portability

Uji kelayakan portability atau portabilitas dalam hal ini adalah pengujian layak digunakan atau tidaknya suatu perangkat lunak dalam hal ini adalah aplikasi *game* apabila dipindahkan atau dijalankan pada perangkat komputer yang berbeda.

3.3.3 Pengujian Sistem

Sebelum diimplementasikan, akan dilakukan peroses pengujian terhadap game simulasi yang telah selesai dibuat dengan menggunakan teknik pengujian perangkat lunak yang telah ada, yaitu:

a) Black-Box

Pengujian dilakukan dengan memperhatikan berbagai fungsi dan kebergunaan dari setiap fungsi atau menu pilihan dalam *game* yang telah dibuat apakah sudah sesuai ataupun belum.

b) White-Box

Pengujian dilakukan dengan cara meneliti dan menganalisa program yang telah dibuat dalam rangka pengecekan kode program apakah ada kesalahan atau tidak, dengan demikian itu akan dilakukan perbaikan.

3.4 Implementasi *Game*

Pembuatan game ini, akan diimplementasikan pada siswa dan siswi jurusan pertanian khususnya yang ada di SMK Negeri 1 Wonosari. Proses pengujian ini bertujuan untuk memberikan sumbangan pemikiran terhadap pembelajaran budidaya tanaman khususnya tanaman jagung. Dalam hal ini juga dapat memberi masukan melalui pengamatan apa saja yang perlu ditambahkan dalam game ini, sehingga kedepannya game ini dapat dikembangkan dan disempurnakan.

BAB IV

HASIL PEMBAHASAN

4.1 Hasil Pengembangan *Game*

4.1.1 Rancangan Umum

Dalam penelitian ini, peneliti merancang sebuah game simulasi yang dimana pada game ini terdapat proses pembelajaran didalamnya, sehingga dapat membantu proses pembelajaran pada siswa.

4.1.2 Rancangan Level

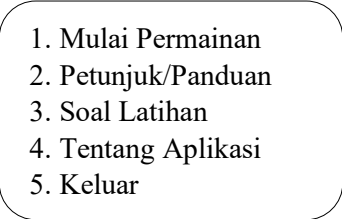
Pada game ini hanya terdapat 1 level saja, namun terdapat beberapa kegiatan – kegiatan. Setiap kegiatan dalam game pemain harus menyelesaikannya dengan waktu yang harus dicapai. Jika kegiatan tidak selesai dan waktu tidak tercapai, maka game akan diulang dari kegiatan awal lagi.

4.1.3 Rancangan Game Budaya Tanaman Jagung

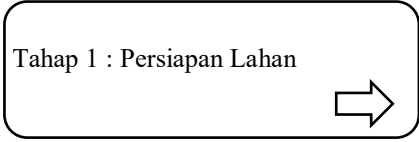
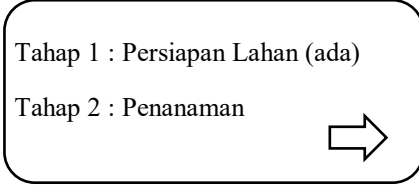
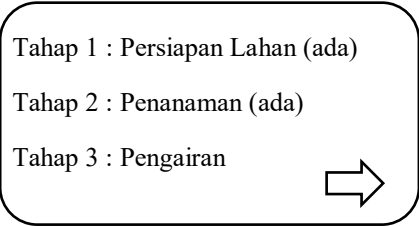
Dalam pembuatan game ini, peneliti merancang sebuah game simulasi dengan memperhatikan parameter-parameter yang sekiranya sangat berpengaruh dalam proses pembelajaran budidaya tanaman jagung itu sendiri. Game akan dimainkan dan diperankan langsung oleh para siswa mulai dari penyiapan alat dan bahan yang diperlukan, proses penyiapan lahan, penanaman benih, perawatan tanaman, serta pemanenan seperti yang terjadi pada kehidupan secara nyata.

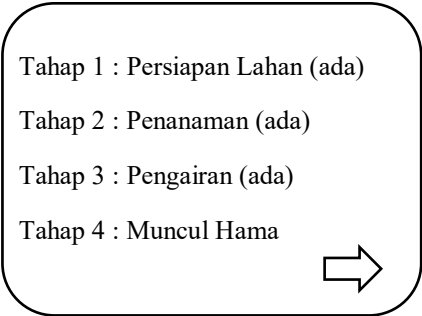
4.1.4 *Storyboard*

Tabel 4.1: *Storyboard* bagian Menu Utama

VISUAL	SKETSA	AUDIO
Menampilkan bagian menu utama dengan beberapa tombol yang memiliki fungsi berbeda.		click.wav (saat klik tombol)

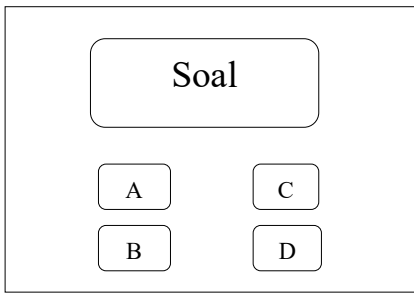
Tabel 4.2: *Storyboard* bagian Mulai Permainan

VISUAL	SKETSA	AUDIO
Menampilkan kegiatan persiapan lahan untuk memulainya proses budidaya tanaman jagung.	 Tahap 1 : Persiapan Lahan	Musik.wav (saat berangsungnya permainan) tuk.wav (saat klik gambar)
Menampilkan kegiatan apa yang harus dilakukan oleh pemain selanjutnya setelah proses persiapan tanaman jagung.	 Tahap 1 : Persiapan Lahan (ada) Tahap 2 : Penanaman	Musik.wav (saat berangsungnya permainan) tuk.wav (saat klik gambar)
Menampilkan kegiatan apa yang harus dilakukan oleh pemain selanjutnya setelah proses persiapan dan penanaman tanaman jagung. Dalam hal ini	 Tahap 1 : Persiapan Lahan (ada) Tahap 2 : Penanaman (ada) Tahap 3 : Pengairan	Musik.wav (saat berangsungnya permainan) tuk.wav

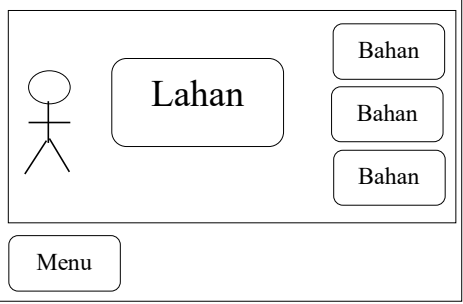
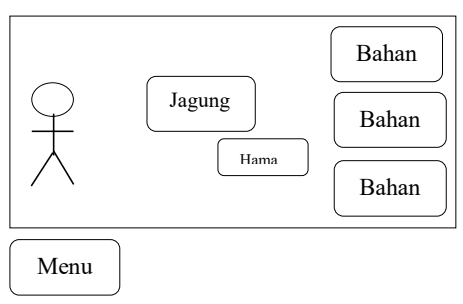
terdapat waktu tertentu yang harus dikejar untuk melanjutkan proses selanjutnya. Jika waktu habis, maka permainan selsai dan akan kembali kepermainan awal.		(saat klik gambar)
Menampilkan kegiatan selanjutnya setelah proses sebelumnya. Dalam hal ini akan muncul hama tanaman serangga yang akan merusak tanaman saat tanaman mulai tumbuh. Dalam hal ini juga terdapat waktu tertentu yang harus dikejar untuk melanjutkan proses selanjutnya.	 Tahap 1 : Persiapan Lahan (ada) Tahap 2 : Penanaman (ada) Tahap 3 : Pengairan (ada) Tahap 4 : Muncul Hama	Musik.wav (saat berangsungnya permainan) Tuk.wav (saat klik gambar)
Menampilkan kegiatan selanjutnya setelah proses sebelumnya. Dalam hal ini adalah proses perawatan tanaman hingga siap panen. Setiap kegiatan perawatan yang	 Tahap 1 : Persiapan Lahan (ada) Tahap 2 : Penanaman (ada) Tahap 3 : Pengairan (ada) Tahap 4 : Muncul Hama (ada) Tahap 5 : Perawatan Tanaman	Musik.wav (saat berangsungnya permainan) tuk.wav

dilakukan juga terdapat waktu tertentu yang harus dikejar untuk melanjutkan keproses selanjutnya. Apabila waktu habis, maka permainan selesai dan akan mengulang dari awal lagi.		(saat klik gambar)
--	--	--------------------

Tabel 4.3: *Storyboard* Soal Latihan

VISUAL	SKETSA	AUDIO
Keterangan : Menampilkan halaman yang berisikan 10 soal latihan dimana proses pengambilan soalnya secara berurutan namun tingkat kesulitannya secara acak. Karakter : Pemain mulai menjawab soal-soal latihan untuk mengukur tingkat pemahaman terhadap pembelajaran dari game.		kuis.wav (saat berlangsungnya kuis dimainkan) win.wav (saat jawaban benar) lose.wav (saat jawaban salah)

Tabel 4.4: Storyboard Sublevel

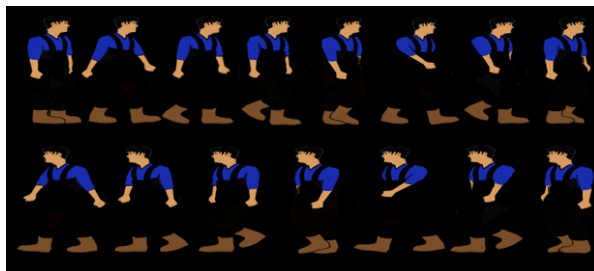
VISUAL	SKETSA	AUDIO
Keterangan : Menampilkan halaman sublevel yang berisi lahan pertanian, bibit tanaman serta alat dan bahan dalam perawatan tanaman. Karakter : Pemain mengatur lahan pertanian kemudian mulai menanam dan merawat tanaman.		Musik.wav (saat berangsungnya permainan) tuk.wav (saat klik gambar)
Keterangan : Menampilkan halaman sublevel yang berisi proses perawatan tanaman dalam membasmi hama-hama yang mengganggu tanaman. Karakter : Pemain mengatur atau memilih bahan-bahan pembasmi hama yang		Musik.wav (saat berangsungnya permainan) tuk.wav (saat klik gambar)

sesuai dengan jenis hama yang ada.		
------------------------------------	--	--

4.1.5 Desain Karakter

a) Aktor

Aktor dalam hal ini adalah siswa yang akan memainkan *game*. Aktor pada game ini terdiri dalam beberapa kegiatan, diantaranya berjalan, mencangkul, menanam, menyiram, memupuk dan menyemprot.



Gambar 4.1: Aktor Berjalan



Gambar 4.2: Aktor Mencangkul



Gambar 4.3: Aktor Menanam



Gambar 4.4: Aktor Menyiram



Gambar 4.5: Aktor Memupuk



Gambar 4.6: Aktor Menyemprot

b) Hama

Hama tanaman jagung yang sering menyerang tanaman jagung pada umumnya adalah dari jenis serangga, dalam hal ini adalah belalang dan ulat yang menyerang pada saat tanaman mulai tumbuh.



Gambar 4.7: Hama Tanaman Jagung Belalang dan Ulat

c) Alat dan Bahan

Cangkul

Digunakan saat menggemburkan tanah dan pembuatan lubang tanam pada proses persiapan penanaman tanaman jagung.



Gambar 4.8: Cangkul

Tanaman Jagung

Tanaman jagung ditanam dengan jarak tanam 70 cm x 20 cm dengan 1 benih/lubang atau 75 cm x 40 cm dengan 2 benih/lubang. Masa pertumbuhannya adalah 80 – 150 hari setelah tanam.



Gambar 4.9: Tanaman Jagung

Air

Air merupakan faktor penting terhadap tumbuh tidaknya tanaman, begipun tanaman jagung sangat membutuhkan air. Pengairan terhadap tanaman jagung dilakukan setiap hari dengan menyesuaikan kelembaban tanah.



Gambar 4.10: Air

Pupuk

Pemupukan pada tanaman jagung sangat penting. Pemupukan dilakukan saat tanaman jagung berumur 7-10 hari setelah tanam yaitu pupuk SP-36, KCI dan $\frac{1}{2}$ bagian Urea dengan cara ditugal atau ditaburkan disamping tanaman. Kemudian pemupukan susulan pada saat tanaman berumur 1 bulan yaitu $\frac{1}{2}$ bagian Urea.



Gambar 4.11: Pupuk Tanaman Jagung

Obat Penyiangan

Penyiangan merupakan proses pembersihan tanaman dari gulma maupun parasit yang mengganggu tanaman jagung. Pada tanaman jagung dilakukan dua kali proses penyiangan, yaitu saat tanaman berumur 2 minggu dan 4 minggu setelah tanam dengan menggunakan obat penyiangan yang disarankan oleh daerah setempat.



Gambar 4.12: Obat Penyiangan

Obat Basmi Hama

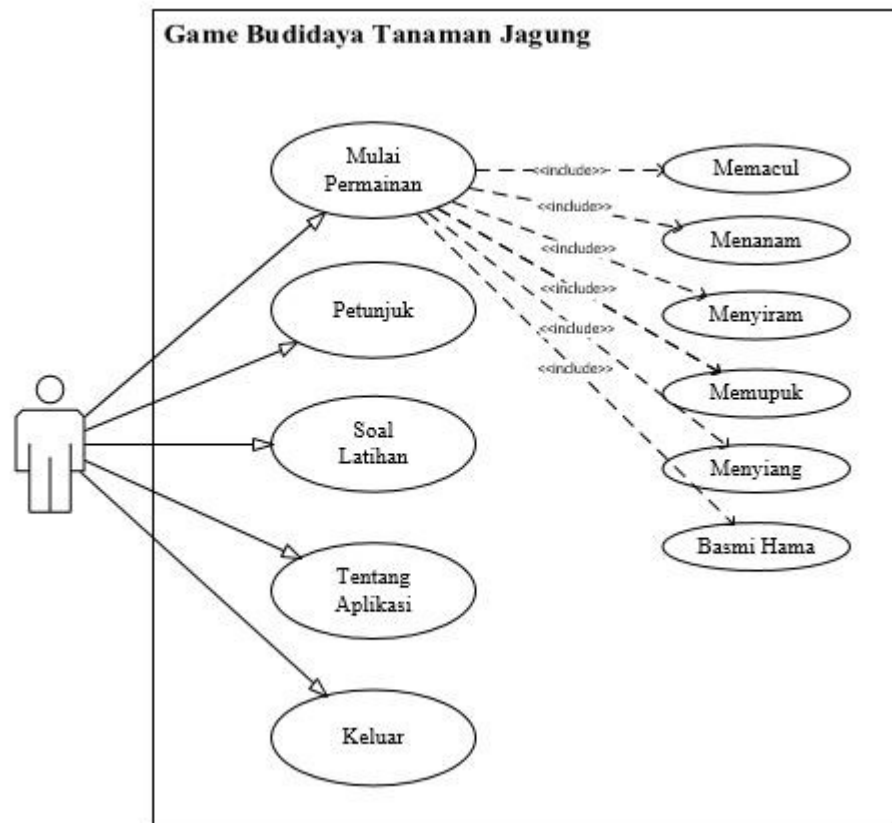
Hama yang sering muncul pada tanaman adalah dari jenis serangga, oleh karena itu obat basmi hama yang sering dianjurkan adalah Insektisida Carbofuran dengan cara menyemprotkan insektisida cair pada tanaman jagung yang mulai diserang oleh hama.



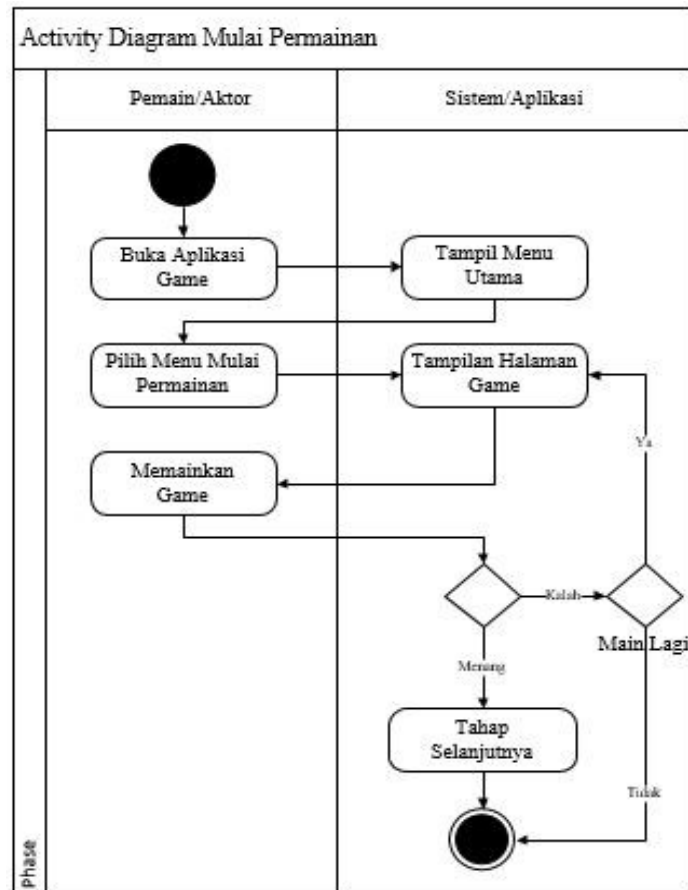
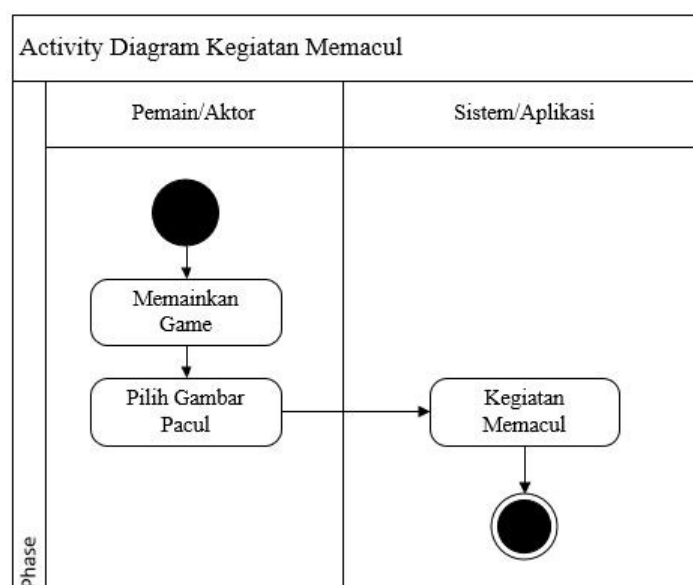
Gambar 4.13: Obat Basmi Hama

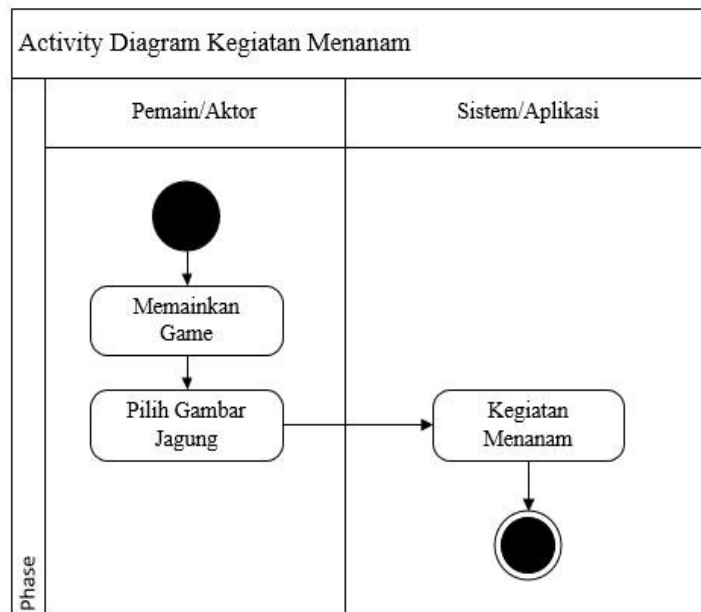
4.1.6 Hasil Analisis Sistem

a) Use Case Diagram Player

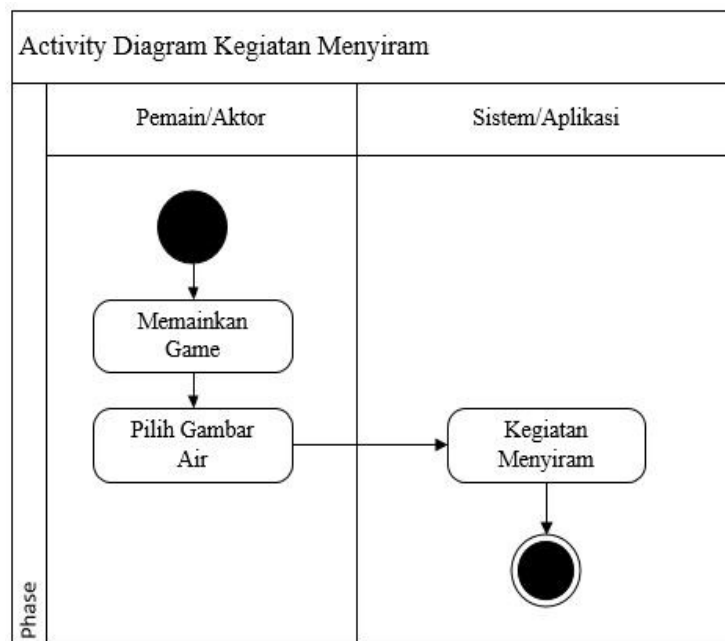


Gambar 4.14: Use Case Diagram Player

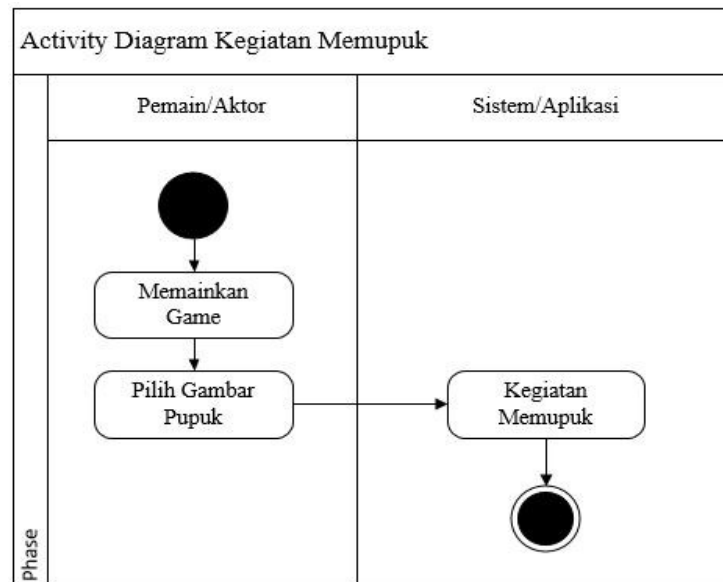
b) *Activity Diagram***Gambar 4.15:** *Activity Diagram* Mulai Permainan**Gambar 4.16:** *Activity Diagram* Kegiatan Memacul



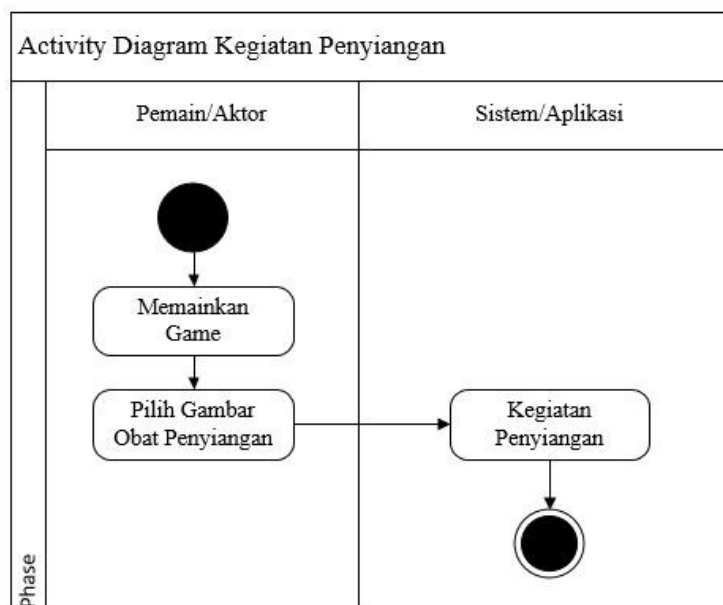
Gambar 4.17: *Activity Diagram Kegiatan Menanam*



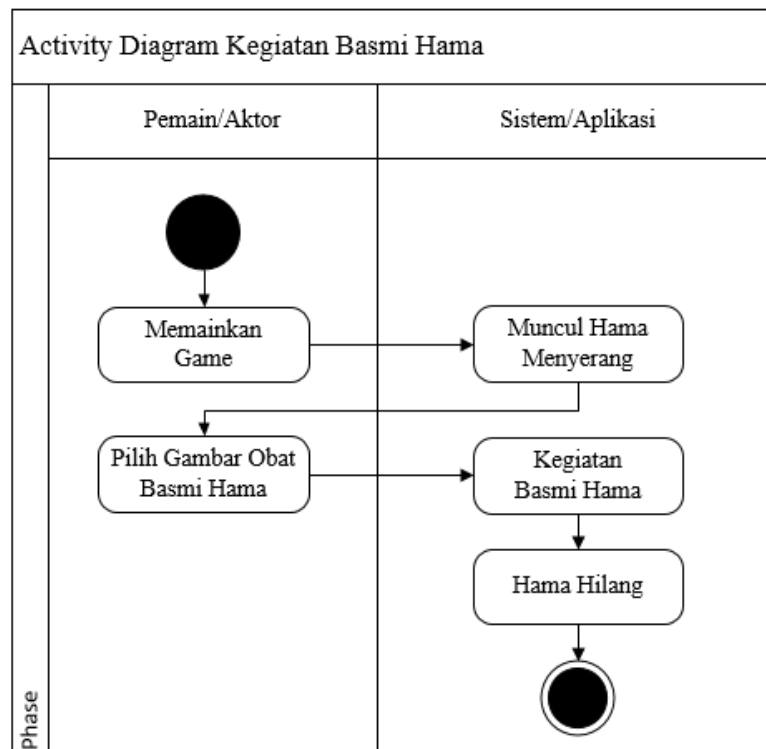
Gambar 4.18: *Activity Diagram Kegiatan Menyiram*



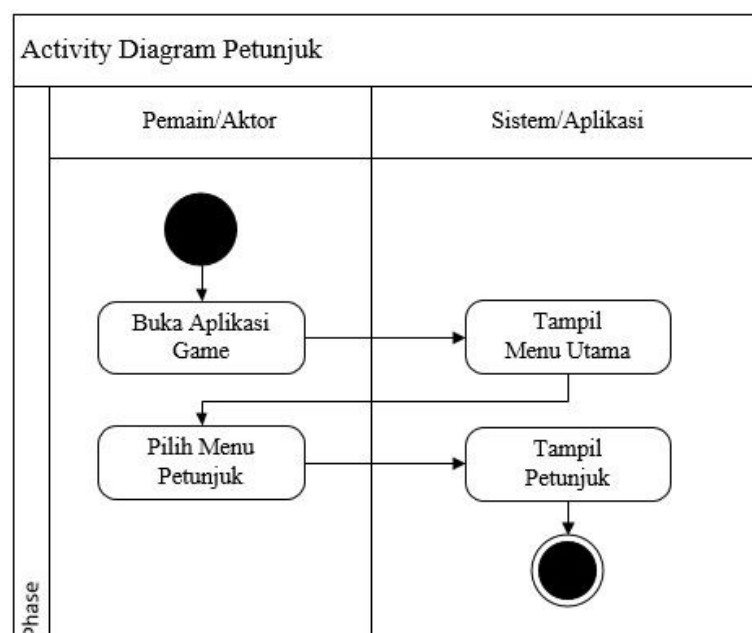
Gambar 4.19: Activity Diagram Kegiatan Memupuk



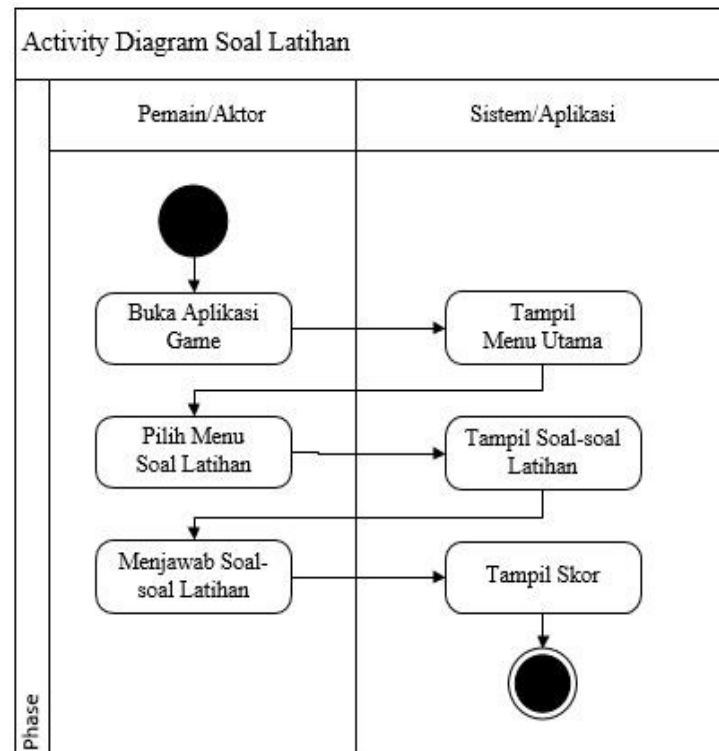
Gambar 4.20: Activity Diagram Kegiatan Penyiangan



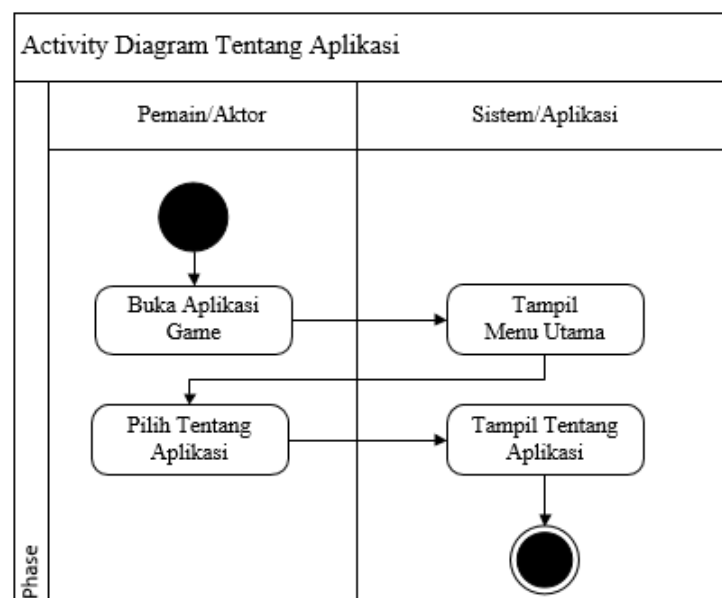
Gambar 4.21: *Activity Diagram* Kegiatan Basmi Hama



Gambar 4.22: *Activity Diagram* Petunjuk

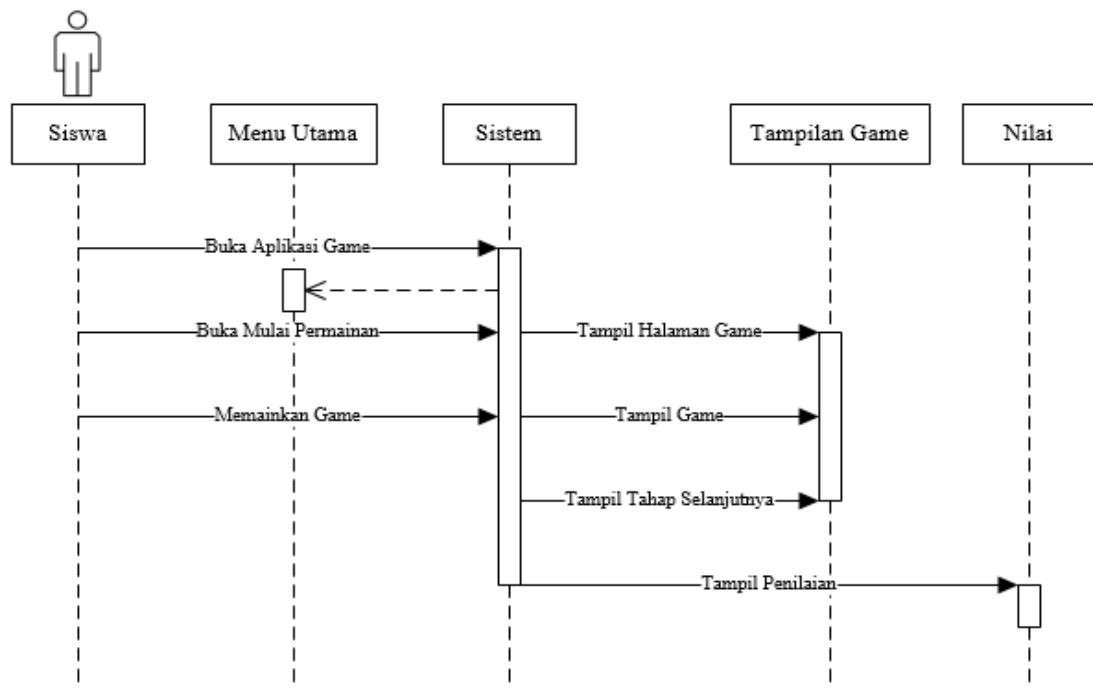


Gambar 4.23: *Activity Diagram Soal Latihan*

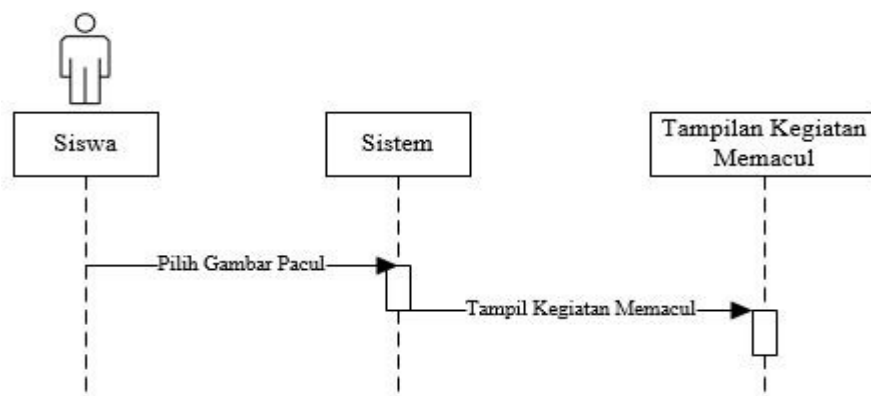


Gambar 4.24: *Activity Diagram Tentang Aplikasi*

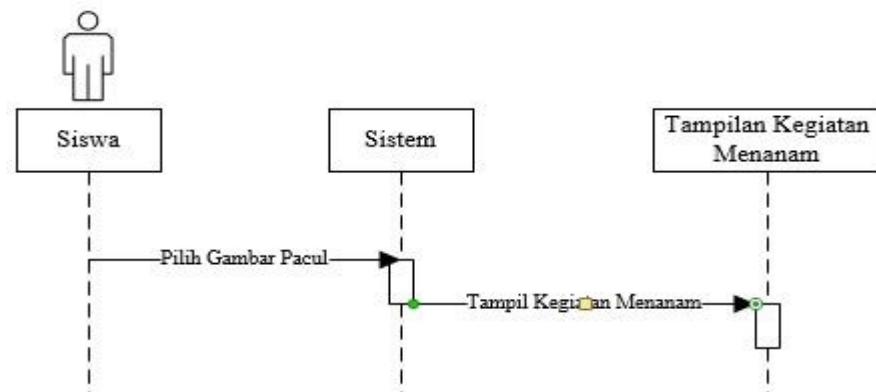
c) *Sequence Diagram*



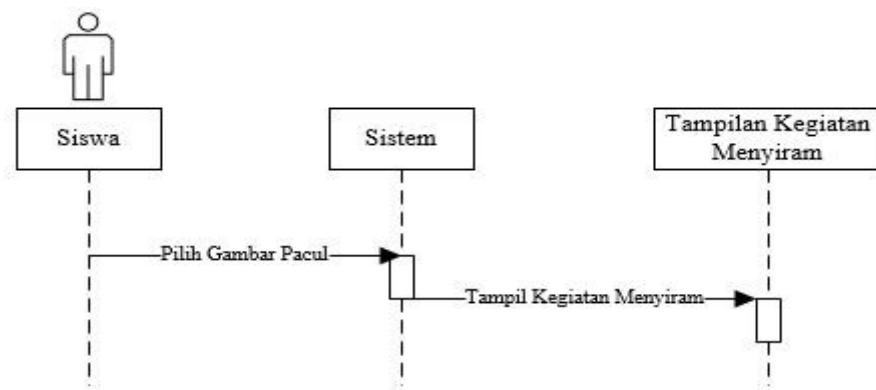
Gambar 4.25: *Sequence Diagram* Mulai Permainan



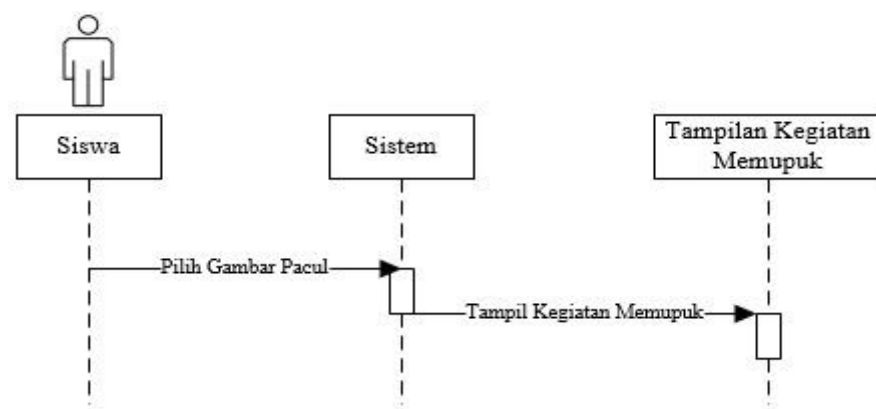
Gambar 4.26: *Sequence Diagram* Kegiatan Memacul



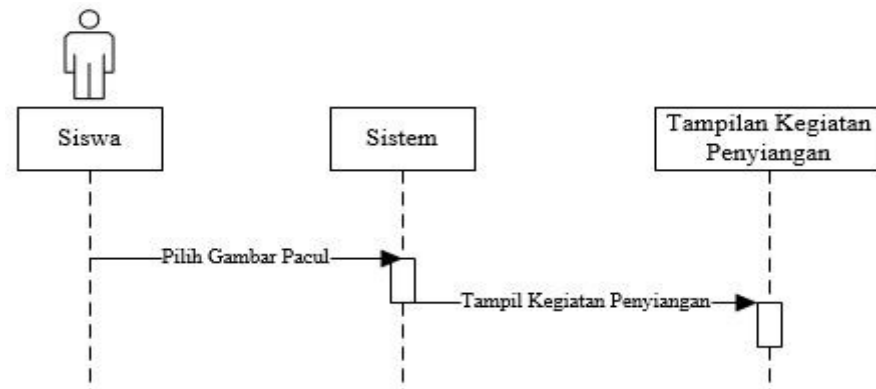
Gambar 4.27: *Sequence Diagram* Kegiatan Menanam



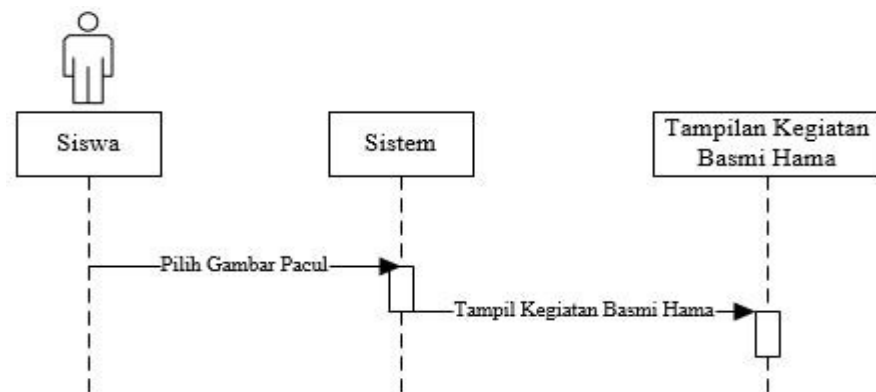
Gambar 4.28: *Sequence Diagram* Kegiatan Menyiram



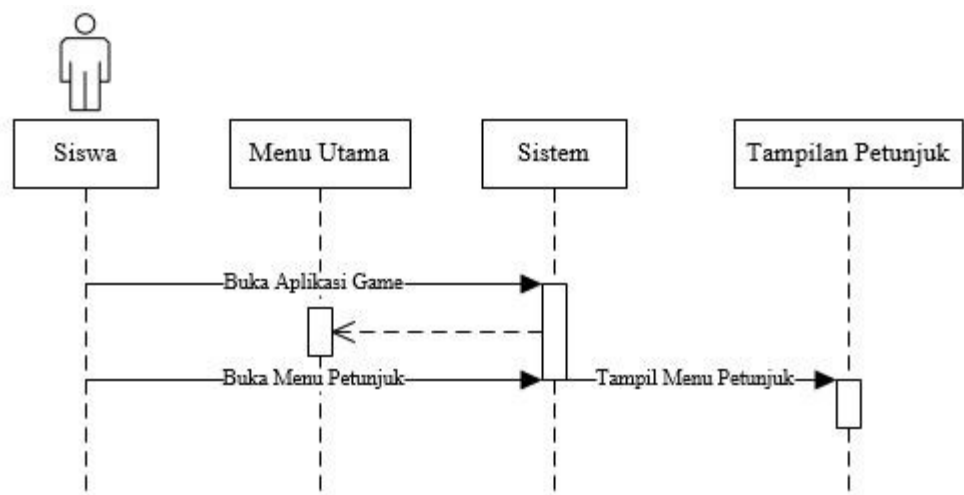
Gambar 4.29: *Sequence Diagram* Kegiatan Memupuk



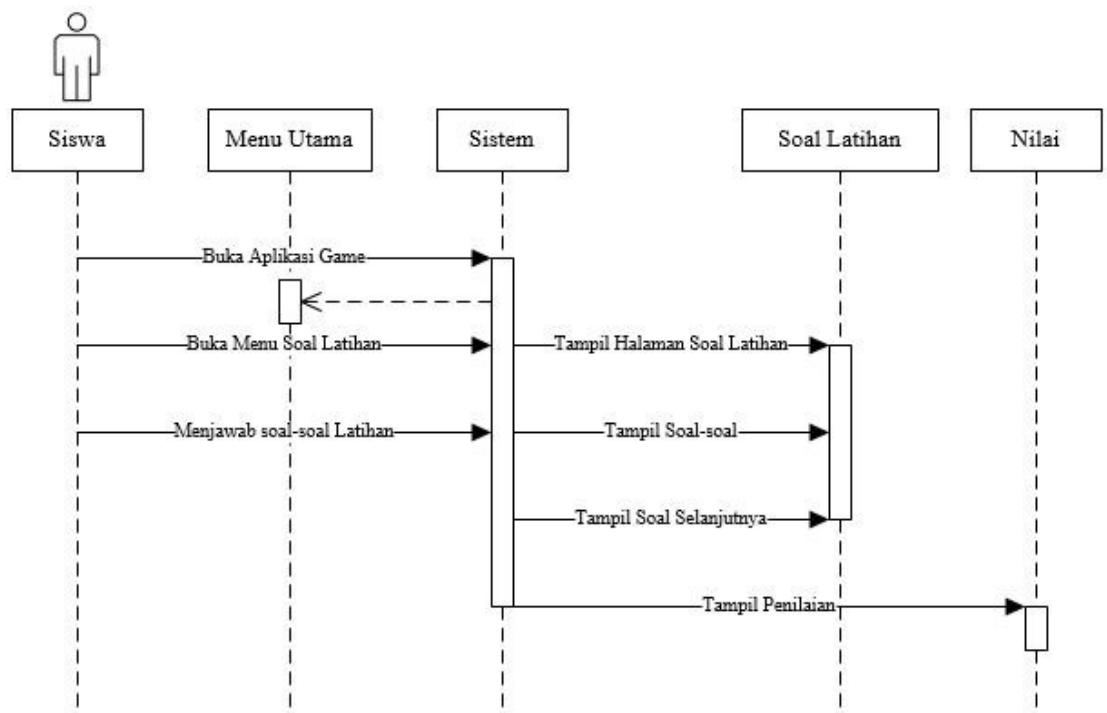
Gambar 4.30: *Sequence Diagram* Kegiatan Penyiangan



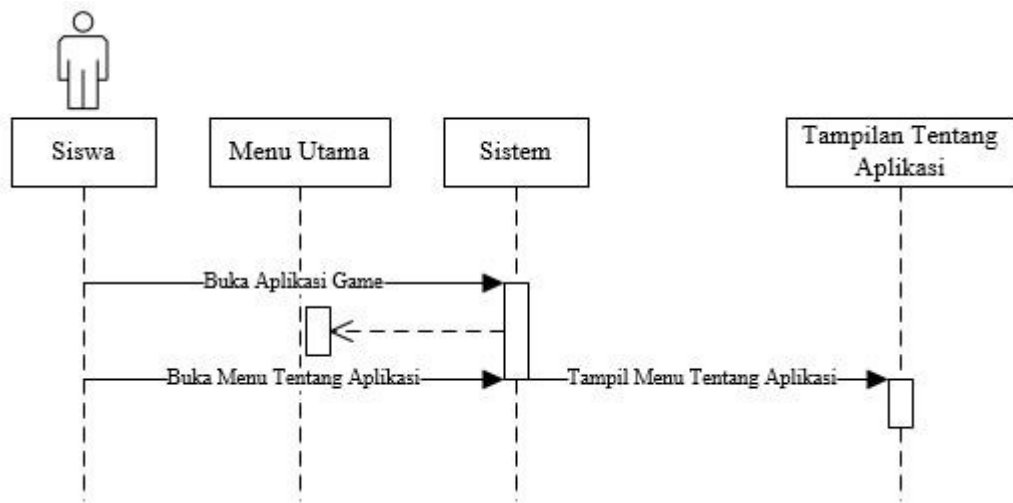
Gambar 4.31: *Sequence Diagram* Kegiatan Basmi Hama



Gambar 4.32: *Sequence Diagram* Petunjuk



Gambar 4.33: *Sequence Diagram* Soal Latihan



Gambar 4.34: *Sequence Diagram* Tentang Aplikasi

4.1.7 Hasil Uji Kelayakan Portability

Berikut merupakan pemindahan aplikasi *game* pada jenis komputer dengan jenis dan kualitas prosesor yang berbeda-beda. Setelah dilakukan penginstalan dan mulai menjalankan *game*, diperoleh hasil uji kelayakan sebagai berikut:

Tabel 4.5: Hasil Uji Kelayakan Portability

No.	Jenis Spek Komputer	Sangat Layak	Layak	Tidak Layak
1	AMD E2-6110 APU with AMD Radeon R2 Graphics 1.50 GHz, RAM 4 GB, 64 bit	√		
2	Intel(R) Core(TM)2 CPU 2.13 GHz RAM 2 GB, 32 bit	√		
3	Genuine Intel(R) CPU 1.60 GHz RAM 2 GB, 32 Bit	√		
4	Intel(R) Pentium(R) CPU 3.10 GHz RAM 4 GB, 64 bit	√		
5	AMD A9-9420 RADEON R5, 5 COMPUTE CORES 2C+3G 3.00 GHz RAM 4 GB, 64 bit	√		

Dari hasil uji kelayakan portability diatas, maka disimpulkan bahwa aplikasi *game* simulasi budidaya tanaman jagung berbasis *real time strategy* dapat digunakan dengan sangat layak pada komputer lainnya.

4.1.8 Hasil Pengujian Sistem

a) Hasil Pengujian Black Box

Tabel 4.6: Hasil Pengujian *Black Box*

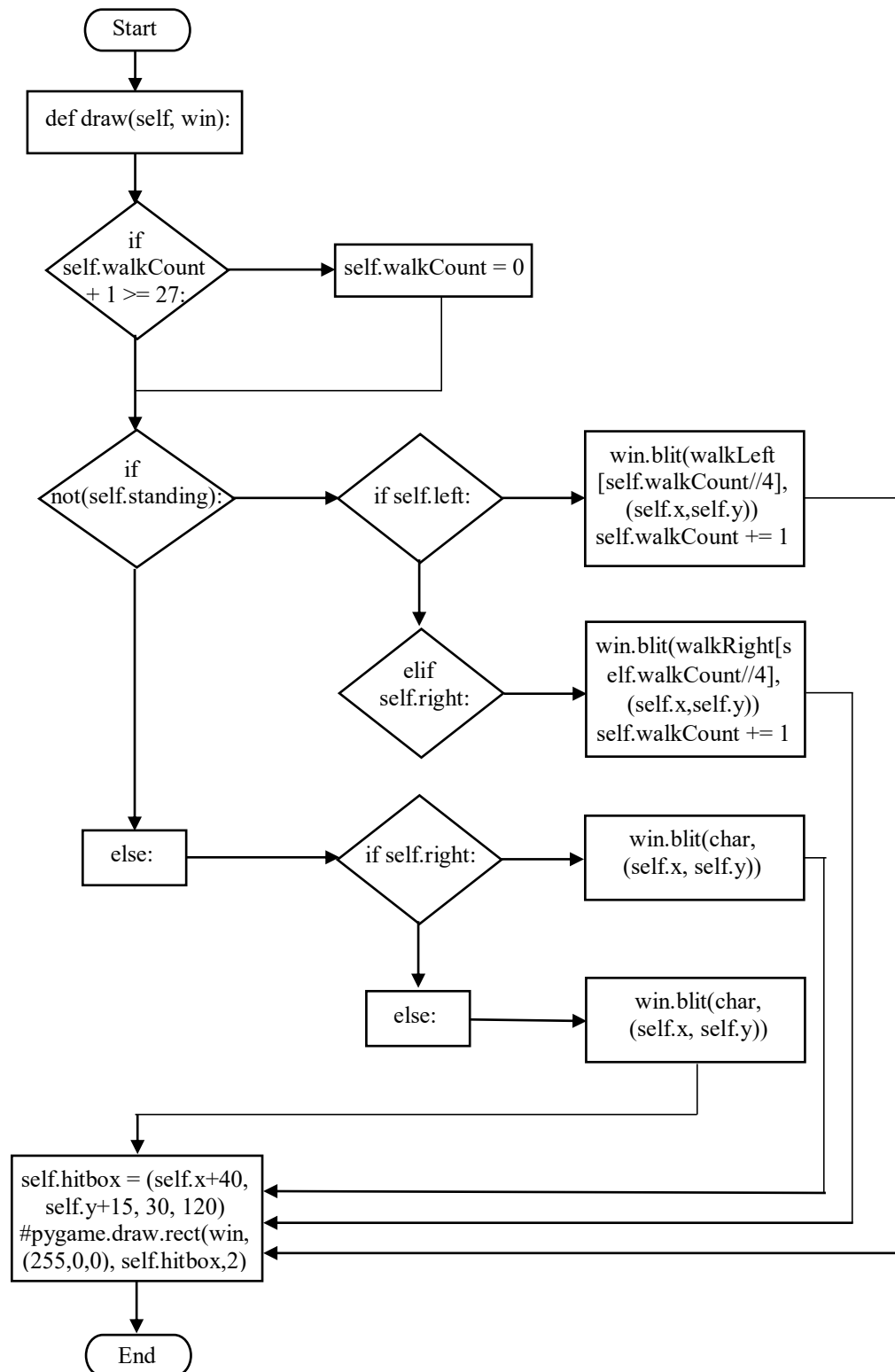
Input/Event	Fungsi	Hasil yang Diharapkan	Hasil Uji
Klik Mulai Permainan	Menampilkan Permainan	Tampilan Permainan	Sukses
Klik Petunjuk	Menampilkan Petunjuk Permainan	Tampilan Petunjuk Permainan	Sukses
Klik Soal Latihan	Menampilkan Soal Latihan	Tampilan Soal Latihan	Sukses
Klik Tentang Aplikasi	Menampilkan Tentang Aplikasi	Tampilan Tentang Aplikasi	Sukses
Klik Keluar	Keluar Dari Permainan	Keluar Permainan	Sukses
Klik Main Lagi	Menampilkan Permainan Awal	Tampilan Permainan	Sukses
Klik Gambar Pacul	Menampilkan Kegiatan Memacul	Tampilan Kegiatan Memacul	Sukses
Klik Gambar Jagung	Menampilkan Kegiatan Menanam	Tampilan Kegiatan Menanam	Sukses
Klik Gambar Air	Menampilkan Kegiatan Menyiram	Tampilan Kegiatan Menyiram	Sukses
Klik Gambar Pupuk	Menampilkan Kegiatan Memupuk	Tampilan Kegiatan Memupuk	Sukses
Klik Gambar Obat Penyiangan	Menampilkan Kegiatan Penyiangan	Tampilan Kegiatan Penyiangan	Sukses
Klik Gambar Obat Basmi Hama	Menampilkan Kegiatan Basmi Hama	Tampilan Kegiatan Basmi Hama	Sukses

Klik Rumah	Gambar	Menampilkan Menu Utama	Tampilan Menu Utama	Sukses
---------------	--------	---------------------------	------------------------	--------

b) Hasil Pengujian White Box

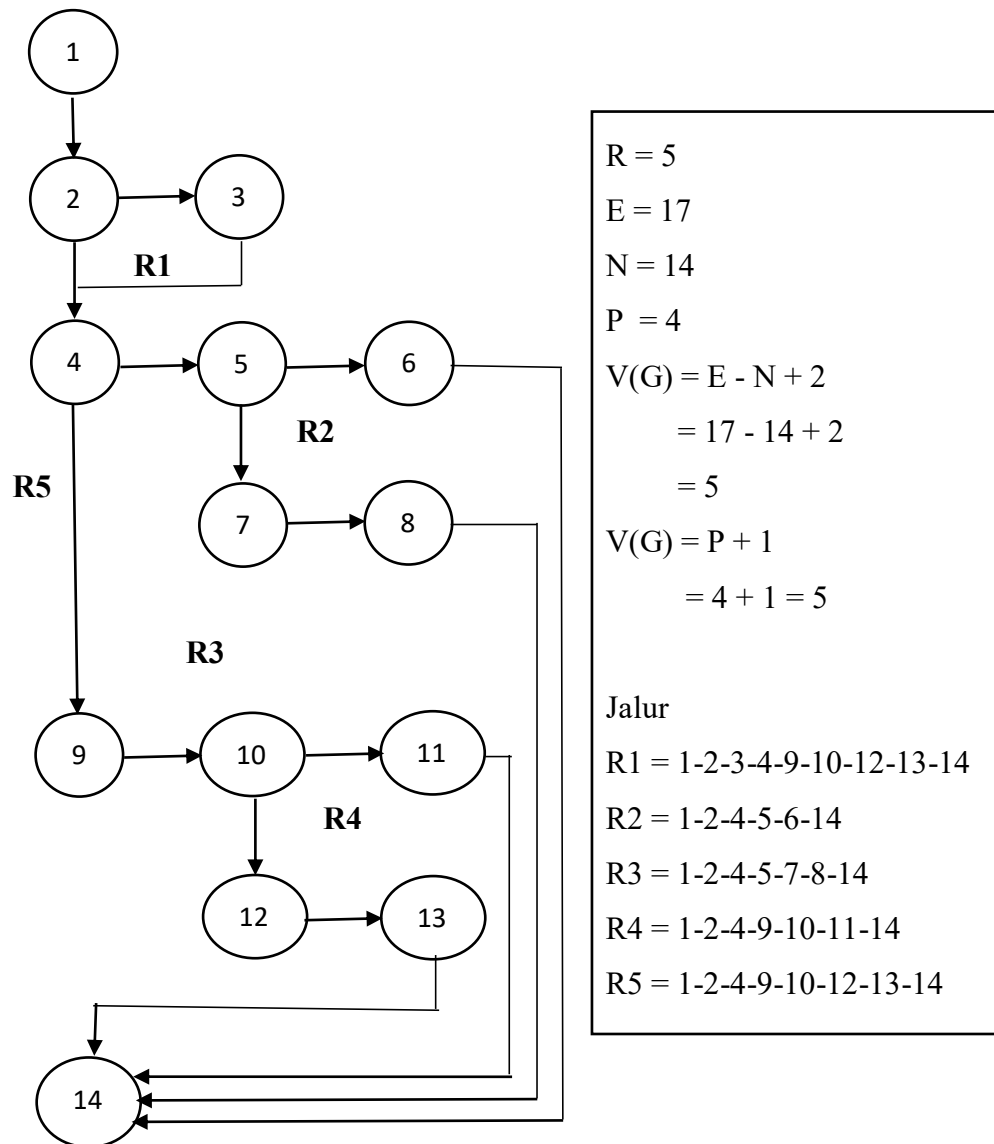
def draw(self, win):	1
if self.walkCount + 1 >= 27:	2
self.walkCount = 0	3
if not(self.standing):	4
if self.left:	5
win.blit(walkLeft[self.walkCount//4], (self.x,self.y))	6
self.walkCount += 1	
elif self.right:	7
win.blit(walkRight[self.walkCount//4], (self.x,self.y))	8
self.walkCount += 1	
else:	9
if self.right:	10
win.blit(char, (self.x, self.y))	11
else:	12
win.blit(char, (self.x, self.y))	13
self.hitbox = (self.x+40, self.y+15, 30, 120)	14
#pygame.draw.rect(win, (255,0,0), self.hitbox,2)	

Flowchart



Gambar 4.35: Flowchart Pengujian White Box

Basic Path



Gambar 4.36: Flowgraph Pengujian White Box

BAB V

PEMBAHASAN

5.1 Pembahasan Model

Penelitian ini menghasilkan sebuah media pembelajaran dan juga game budidaya tanaman jagung yang sekiranya dapat membantu siswa dalam proses pembelajaran budidaya tanaman jagung.

5.2 Pembahasan Sistem

5.2.1 Tampilan Menu Utama



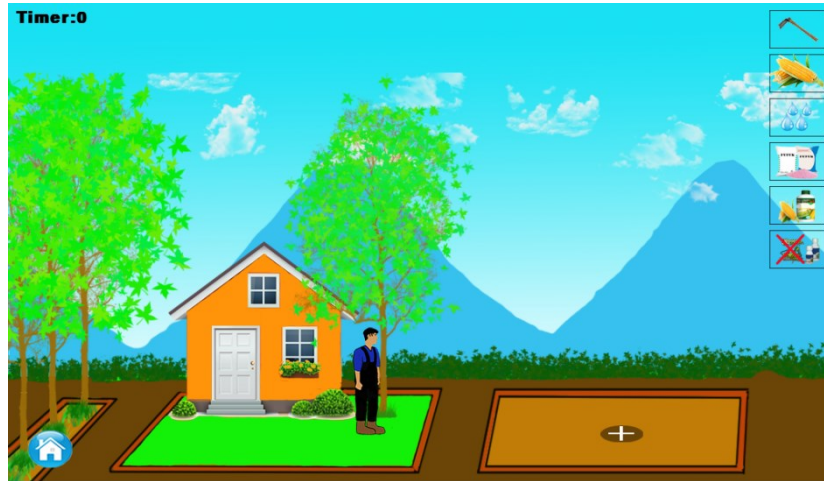
Gambar 5.1: Tampilan Menu Utama

Tampilan ini terdapat beberapa menu antara lain :

1. Mulai Permainan : untuk memulai permainan.
2. Petunjuk : berisikan petunjuk untuk mengetahui bagaimana cara bermain.
3. Soal Latihan : berisikan soal-soal latihan untuk menguji pemahaman pemain terhadap budidaya tanaman jagung dari game.
4. Tentang Aplikasi : berisikan profil tentang penulis dan tentang tujuan pembuatan game.
5. Keluar : untuk keluar dari permainan.

5.2.2 Tampilan Mulai Permainan

Halaman ini berisikan tampilan dimana pemain akan memulai awal dari proses pembudidayaan tanaman jagung.



Gambar 5.2: Tampilan Mulai Permainan

Tampilan ini terdapat beberapa gambar yang memiliki fungsi masing-masing diantaranya :

1. Gambar Pacul : memacul tanah untuk lubang tanam.
2. Gambar Jagung : benih tanaman jagung.
3. Gambar Air : untuk menyiram tanaman.
4. Gambar Pupuk : untuk memupuk tanaman.
5. Gambar Obat Penyiangan : untuk penyiangan tanaman.
6. Gambar Obat Anti Hama : untuk membasmi hama yang merusak tanaman.
7. Gambar Halaman Rumah : untuk kembali ke menu utama.

5.2.3 Tampilan Kegiatan Mamacul Untuk Persiapan Penanaman

Halamain ini berisikan tampilan dimana pemain akan memulai proses pengolahan tanah untuk penanaman tanaman jagung.



Gambar 5.3: Tampilan Kegiatan Mamacul

5.2.4 Tampilan Kegiatan Penanaman Tanaman Jagung

Halamain ini berisikan tampilan dimana pemain akan memulai proses penanaman benih tanaman jagung.



Gambar 5.4: Tampilan Kegiatan Penanaman Tanaman Jagung

5.2.5 Tampilan Kegiatan Penyiraman Tanaman

Halamain ini berisikan tampilan dimana pemain melakukan kegiatan penyiraman tanaman jagung pada tahap pertumbuhan awal.



Gambar 5.5: Tampilan Kegiatan Penyiraman Tanaman

5.2.6 Tampilan Kegiatan Pemupukan Tanaman

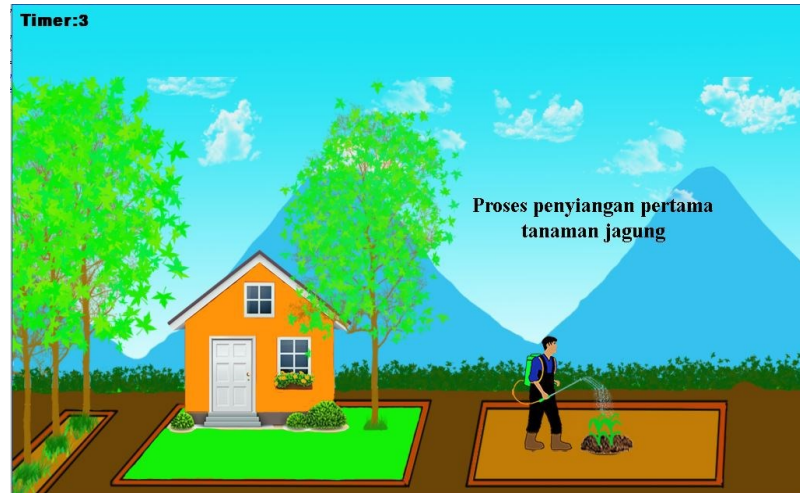
Halaman ini berisikan tampilan dimana pemain melakukan kegiatan pemupukan tanaman jagung pertama.



Gambar 5.6: Tampilan Kegiatan Pemupukan Pertama

5.2.7 Tampilan Kegiatan Penyiangan Tanaman

Halamain ini berisikan tampilan dimana pemain melakukan kegiatan penyiangan pertama tanaman jagung.



Gambar 5.7: Tampilan Kegiatan Penyiangan Pertama

5.2.8 Tampilan Kegiatan Pembasmian Hama

Halamain ini berisikan tampilan dimana pemain melakukan kegiatan pengendalian hama pada tanaman jagung.



Gambar 5.8: Tampilan Kegiatan Pengendalian Hama

5.2.9 Tampilan Menu Petunjuk

Halaman ini berisikan tentang petunjuk dalam permainan budidaya tanaman jagung, seperti pada gambar dibawah ini :



Gambar 5.9: Tampilan Petunjuk Permainan

5.2.10 Tampilan Menu Soal Latihan

Halaman ini berisikan soal-soal latihan untuk menguji pemahaman pemain terhadap budidaya tanaman jagung setelah bermain game, seperti pada gambar dibawah ini :



Gambar 5.10: Tampilan Soal Latihan

5.2.11 Tampilan Menu Tentang Aplikasi

Halaman ini berisikan profil dari pembuat game dan keterangan tentang game budidaya tanaman jagung, seperti pada gambar dibawah ini :



Gambar 5.11: Tampilan Tentang Aplikasi

BAB VI

PENUTUP

6.1 Kesimpulan

Berdasarkan tahapan-tahapan yang telah penulis lakukan maka dapat disimpulkan bahwa penelitian ini menghasilkan sebuah aplikasi *Game* Simulasi Pembelajaran Budidaya Tanaman Jagung Berbasis *Real Time Strategi* yang memberikan pembelajaran dan pengetahuan terhadap budidaya tanaman jagung.

6.2 Saran

Adapun saran dari penulis kepada peneliti selanjutnya khususnya dalam pengembangan *game* Simulasi Pembelajaran Budidaya Tanaman Jagung Berbasis *Real Time Strategi* ini sebagai berikut :

1. Adanya perubahan cuaca seperti siang, malam, musim panas dan musim ujan agar lebih menambah kesan seperti pada kehidupan secara nyata.
2. Penambahan terhadap level dan jumlah tanaman serta alat dan bahan perlengkapan yang sekiranya dibutuhkan dalam budidaya tanaman jagung.
3. Penambahan terhadap penyakit dan hama lainnya yang terdapat pada tanaman jagung.

DAFTAR PUSTAKA

- [1] V. E. J. Mawitjere, “DAYA SAINS : Jurnal Pengabdian Kepada Masyarakat Menanamkan Jiwa Kewirausahaan melalui Percontohan Budidaya Tanaman Pangan pada Anak Usia Sekolah,” vol. 1, 2018.
- [2] C. Hanum, *Teknik Budidata Tanaman Jilid 2 untuk Sekolah Menengah kejuruan*, vol. 53, no. 9. 2008.
- [3] Haryanto, K. R. Purba, and K. Gunadi, “Pembuatan Strategy Farming Game Berbasis Flash,” *J. Infra*, vol. 4, no. 1, pp. 1–7, 2016.
- [4] S. Rostianingsih, S. B. Gregorius, and H. K. Wijaya, “Game Simulasi Finite State Machine Untuk Pertanian Dan Peternakan,” *J. DKV Adiwarna*, vol. 5, pp. 2–7, 2013.
- [5] N. R. Della, I. D. Wijaya, and B. Harijanto, “Pengembangan Game Simulasi Penanaman Padi Situ Bagendit (Studi Kasus Pada Uptd Pertanian Kecamatan Ngantang),” *J. Inform. Polinema*, vol. 3, no. 4, pp. 29–35, 2017.
- [6] A. D. Wulandari, “Game Edukatif Sejarah Komputer Menggunakan Role Playing Game (RPG) MAKER XP Sebagai Media Pembelajaran di SMP Negeri 2 Kalibawang,” *Universitas Negeri Yogyakarta*, vol. Unknown, no. Unknown, p. No Pages, 2012.
- [7] W. Pratama, “Game Adventure Misteri Kotak Pandora,” *J. Telemat.*, vol. 7, no. 2, pp. 13–31, 2014.
- [8] C. D. Irawan, D. J. Mamahit, and A. M. Sambul, “Pembuatan Game Simulasi Kewirausahaan untuk Profesi Petani,” *J. Tek. Inform.*, vol. 14, no. 1, pp. 71–78, 2019.
- [9] E. S. Kw, S. Rostianingsih, and K. Gunadi, “Pembuatan Game Simulasi Perkebunan , Peternakan dan Interaksi Sosial,” pp. 1–4, 2013.
- [10] D. Novaliendry, “APLIKASI GAME GEOGRAFI BERBASIS MULTIMEDIA INTERAKTIF (STUDI KASUS SISWA KELAS IX SMPN 1 RAO),” *Fordham Law Rev.*, vol. 79, no. 6, pp. 2395–2424, 2013.
- [11] Riwandi, M. Handajaningsih, and Hasanudin, *TEKNIK BUDIDAYA*

JAGUNG DENGAN SISTEM ORGANIK DI LAHAN MARJINAL. 2014.

- [12] F. Muhadjir, "Budidaya Tanaman Jagung," *Badan Penelit. dan Pengemb. Pertanian. Bogor*. 423 hal, p. 21, 2009.
- [13] W. Aprianti and U. Maliha, "Sistem Informasi Kepadatan Penduduk Kelurahan Atau Desa Studi Kasus Pada Kecamatan Bati-Bati," vol. 2, no. 2013, pp. 21–28, 2016.
- [14] A. Hendini, "PEMODELAN UML SISTEM INFORMASI MONITORING PENJUALAN DAN STOK BARANG (STUDI KASUS DITRO ZHEZHA PONTIANAK)," *Crop Sci.*, vol. 23, no. 2, p. 201, 2016.
- [15] Y. P. W. Simaremare, A. Pribadi, and R. P. Wibowo, "Perancangan dan Pembuatan Aplikasi Manajemen Publikasi Ilmiah Berbasis Online pada Jurnal SISFO," *Tek. Pomits*, vol. 2, no. 3, pp. 470–475, 2013.
- [16] N. Azwanti, "SISTEM INFORMASI PENJUALAN TAS BERBASIS WEB DENGAN PEMODELAN UML," vol. v, no. 6, pp. 2827–2839, 2017.
- [17] E. Supriyo, "Sistem Informasi Desa Lalang Sembawa Banyuasin," *J. Inform.*, vol. 1, p. 42, 2015.
- [18] M. S. Dewi, "Penggunaan Aplikasi Adobe Photoshop Dalam Meningkatkan Keterampilan Editing Foto Bagi Anak Tunarungu," *Ann. Neurol.*, vol. 23, no. 1 S, pp. S88–S91, 2012.
- [19] I. B. Trisno, *Belajar Pemrograman Sulit ? Coba Python*. 2016.
- [20] I. B. Muktyas and S. Arifin, "Semua Subgrup Siklus Dari Grup (Zn,+)," vol. 3, no. 2, pp. 177–186, 2018.
- [21] Mulya Megah & Sukemi, "Pengujian DII Untuk Meningkatkan Modularitas Dan Portabilitas Perangkat Lunak Yang Dikembangkan Dengan Paradigma Berorientasi Objek," vol. 4, no. 3, 2009.
- [22] M. Komarudin MZ, "Pengujian perangkat Lunak metode Black box berbasis partitions pada aplikasi sistem informasi di sekolah," *J. Mikrotik*, vol. o6, no. 3, pp. 02–16, 2016.
- [23] M. S. Mustaqbal, R. F. Firdaus, and H. Rahmadi, "PENGUJIAN APLIKASI MENGGUNAKAN BLACK BOX TESTING BOUNDARY VALUE ANALYSIS (Studi Kasus : Aplikasi Prediksi Kelulusan SNMPTN)," vol. I,

no. 3, pp. 31–36, 2015.

KODE PROGRAM

Menu Utama

```
import pygame, sys
import os
from tkinter import *
from tkinter import messagebox
from pygame.locals import *
pygame.init()

pygame.display.set_caption('Menu Utama')
mainClock = pygame.time.Clock()
win = pygame.display.set_mode((1024, 600),0,32)
bg = pygame.image.load(os.path.join('menu', 'MU.png'))
font = pygame.font.SysFont(None, 20)
sounds = {"click":pygame.mixer.Sound("musik/click.wav")}]

def main_menu():
    while True:
        win.blit(bg, (0,0))
        img1 = pygame.image.load(os.path.join('menu', 'MP.png'))
        img2 = pygame.image.load(os.path.join('menu', 'P.png'))
        img3 = pygame.image.load(os.path.join('menu', 'S.png'))
        img4 = pygame.image.load(os.path.join('menu', 'TAP.png'))
        img5 = pygame.image.load(os.path.join('menu', 'KP.png'))

        mx, my = pygame.mouse.get_pos()
        button_1 = pygame.Rect(398, 220, 250, 50)
        button_2 = pygame.Rect(398, 280, 250, 50)
        button_3 = pygame.Rect(398, 340, 250, 50)
        button_4 = pygame.Rect(398, 400, 250, 50)
        button_5 = pygame.Rect(398, 460, 250, 50)

        if button_1.collidepoint((mx, my)):
            if click:
                sounds["click"].play()
                game()
        if button_2.collidepoint((mx, my)):
            if click:
                sounds["click"].play()
                aturan()
        if button_3.collidepoint((mx, my)):
            if click:
                sounds["click"].play()
                latihan()
        if button_4.collidepoint((mx, my)):
            if click:
                sounds["click"].play()
```

```

        tentang()
    if button_5.collidepoint((mx, my)):
        if click:
            sounds["click"].play()
            pygame.quit()
            sys.exit()
    #pygame.draw.rect(win, (255, 0, 0), button_1)
    win.blit(img1, button_1)
    #pygame.draw.rect(win, (255, 0, 0), button_2)
    win.blit(img2, button_2)
    #pygame.draw.rect(win, (255, 0, 0), button_3)
    win.blit(img3, button_3)
    #pygame.draw.rect(win, (255, 0, 0), button_4)
    win.blit(img4, button_4)
    #pygame.draw.rect(win, (255, 0, 0), button_5)
    win.blit(img5, button_5)

    click = False
    for event in pygame.event.get():
        if event.type == QUIT:
            pygame.quit()
            sys.exit()
        if event.type == KEYDOWN:
            if event.key == K_ESCAPE:
                pygame.quit()
                sys.exit()
        if event.type == MOUSEBUTTONDOWN:
            if event.button == 1:
                pygame.time.delay(500)
                click = True
    pygame.display.update()
    mainClock.tick(60)

def game():
    running = True
    while running:
        import game
        game.game()
        for event in pygame.event.get():
            if event.type == QUIT:
                pygame.quit()
                sys.exit()
            if event.type == KEYDOWN:
                if event.key == K_ESCAPE:
                    running = True
        pygame.display.update()
        mainClock.tick(60)

def aturan():
    running = True
    while running:

```

```

pygame.display.set_caption('Petunjuk')
mainClock = pygame.time.Clock()
win = pygame.display.set_mode((1024, 600), 0, 32)
bg = pygame.image.load(os.path.join('menu', 'PP.png'))
font = pygame.font.SysFont(None, 20)
sounds = {"click": pygame.mixer.Sound("Musik/click.wav")}

def at():
    while True:
        win.blit(bg, (0, 0))
        img1 = pygame.image.load(os.path.join('menu', 'awal.png'))

        mx, my = pygame.mouse.get_pos()
        button_1 = pygame.Rect(827, 405, 70, 60)

        if button_1.collidepoint((mx, my)):
            if click:
                sounds["click"].play()
                game()
            #pygame.draw.rect(win, (255, 0, 0), button_1)
            win.blit(img1, button_1)

        click = False
        for event in pygame.event.get():
            if event.type == QUIT:
                pygame.quit()
                sys.exit()
            if event.type == KEYDOWN:
                if event.key == K_ESCAPE:
                    pygame.quit()
                    sys.exit()
            if event.type == MOUSEBUTTONDOWN:
                if event.button == 1:
                    pygame.time.delay(500)
                    click = True
        pygame.display.update()
        mainClock.tick(60)

def game():
    running = True
    while running:
        main_menu()
        for event in pygame.event.get():
            if event.type == QUIT:
                pygame.quit()
                sys.exit()
            if event.type == KEYDOWN:
                if event.key == K_ESCAPE:
                    running = False
        pygame.display.update()

```

```

        mainClock.tick(60)
    at()

    for event in pygame.event.get():
        if event.type == QUIT:
            pygame.quit()
            sys.exit()
        if event.type == KEYDOWN:
            if event.key == K_ESCAPE:
                running = False
    pygame.display.update()
    mainClock.tick(60)

def latihan():
    running = True
    while running:
        global q,score
        q = 1
        score = 0

        def wrongAns():
            Tk().wm_withdraw()
            messagebox.showinfo('Peringatan!!', 'Jawaban Anda Salah!!!')
        def rightAns():
            Tk().wm_withdraw()
            messagebox.showinfo('Peringatan!!', 'Jawaban Anda Benar')
            global score
            score += 1

        def checkAns(mousePos):
            if (mousePos[0] in range(160, 420)) and (mousePos[1] in range(410, 482)):
                global q
                #pilihan A
                if q == 3 or q == 4 or q == 6:
                    correctAudio.play()
                    rightAns()
                else:
                    wrongAudio.play()
                    wrongAns()
                q+=1

            elif (mousePos[0] in range(588, 847)) and (mousePos[1] in range(410, 482)):
                #pilihan C
                if q == 1 or q == 8:
                    correctAudio.play()
                    rightAns()
                else:
                    wrongAudio.play()
                    wrongAns()
                q+=1

```

```

elif (mousePos[0] in range(588, 847)) and (mousePos[1] in range(410, 552)):
    #pilihan D
    if q == 7 or q == 10:
        correctAudio.play()
        rightAns()
    else:
        wrongAudio.play()
        wrongAns()
    q+=1

```

```

elif (mousePos[0] in range(160, 490)) and (mousePos[1] in range(410, 552)):
    #pilihan B
    if q == 2 or q == 5 or q == 9:
        correctAudio.play()
        rightAns()
    else:
        wrongAudio.play()
        wrongAns()
    q+=1

```

2 – Initialize the game

```

pygame.init()
pygame.mixer.init()
wrongAudio = pygame.mixer.Sound("sound/lose.wav")
correctAudio = pygame.mixer.Sound("sound/win.wav")
width, height = 1024, 600
screen=pygame.display.set_mode((width, height))

```

3 – Load images

```

bgImage = pygame.image.load("kuis/latihan.png")
question1 = pygame.image.load("kuis/soal1.png")
question2 = pygame.image.load("kuis/soal2.png")
question3 = pygame.image.load("kuis/soal3.png")
question4 = pygame.image.load("kuis/soal4.png")
question5 = pygame.image.load("kuis/soal5.png")
question6 = pygame.image.load("kuis/soal6.png")
question7 = pygame.image.load("kuis/soal7.png")
question8 = pygame.image.load("kuis/soal8.png")
question9 = pygame.image.load("kuis/soal9.png")
question10 = pygame.image.load("kuis/soal10.png")

```

```

screen.blit(bgImage, (0, 0))
pygame.display.flip()
Tk().wm_withdraw() # to hide the main window
messagebox.showinfo("Selesaikan Quis", "Mulai QUIIS")

```

4 – Keep looping through

```

while 1:
    # 5 – Clear the screen before drawing it again
    screen.fill(0)
    # 6 - Draw the screen elements

```

```

if q == 1:
    screen.blit(question1, (0, 0))
elif q == 2:
    screen.blit(question2, (0, 0))
elif q == 3:
    screen.blit(question3, (0, 0))
elif q == 4:
    screen.blit(question4, (0, 0))
elif q == 5:
    screen.blit(question5, (0, 0))
elif q == 6:
    screen.blit(question6, (0, 0))
elif q == 7:
    screen.blit(question7, (0, 0))
elif q == 8:
    screen.blit(question8, (0, 0))
elif q == 9:
    screen.blit(question9, (0, 0))
elif q == 10:
    screen.blit(question10, (0, 0))
elif q > 10:

    if score > 6:
        finalMessage = " Skor kamu adalah " + str(score) + " dari 10 Soal. " + " Kembali Ke
Menu Utama "
        Tk().wm_withdraw() # to hide the main window
        messagebox.showinfo("Pengetahuan kamu luar biasa", finalMessage)
        score = 0
        main_menu()

    elif score <= 6:
        finalMessage = " Skor kamu adalah " + str(score) + " dari 10 soal. " + " Kembali Ke
Menu Utama "
        Tk().wm_withdraw() # to hide the main window
        messagebox.showinfo("pengetahuan kamu kurang baik ", finalMessage)
        score = 0
        main_menu()

# 7 – Update the screen
pygame.display.flip()
# 8 – Loop through the events
for event in pygame.event.get():
    # Check if the event is the X button
    if event.type==pygame.QUIT:

# If it is quit the game
    Tk().wm_withdraw() # to hide the main window
    messagebox.showinfo('Peringatan!!', 'Apakah Anda Menyerah ??')
    pygame.quit()
    exit(0)

```

```

        if event.type == pygame.MOUSEBUTTONDOWN:
            mousePos = pygame.mouse.get_pos()
            checkAns(mousePos)
            break
    exit(0)

def tentang():
    running = True
    while running:
        pygame.time.delay(1000)
        pygame.display.set_caption('Tentang Aplikasi')
        mainClock = pygame.time.Clock()
        win = pygame.display.set_mode((1024, 600), 0, 32)
        bg = pygame.image.load(os.path.join('menu', 'TA.png'))
        font = pygame.font.SysFont(None, 20)
        sounds = {"click": pygame.mixer.Sound("Musik/click.wav")}

    def ta():
        while True:
            win.blit(bg, (0, 0))
            img1 = pygame.image.load(os.path.join('menu', 'awal.png'))
            mx, my = pygame.mouse.get_pos()

            button_1 = pygame.Rect(827, 405, 70, 60)

            if button_1.collidepoint((mx, my)):
                if click:
                    sounds["click"].play()
                    game()
            #pygame.draw.rect(win, (255, 0, 0), button_1)
            win.blit(img1, button_1)

            click = False
            for event in pygame.event.get():
                if event.type == QUIT:
                    pygame.quit()
                    sys.exit()
                if event.type == KEYDOWN:
                    if event.key == K_ESCAPE:
                        pygame.quit()
                        sys.exit()
                if event.type == MOUSEBUTTONDOWN:
                    if event.button == 1:
                        pygame.time.delay(500)
                        click = True
            pygame.display.update()
            mainClock.tick(60)

    def game():
        running = True
        while running:

```

```

        main_menu()
    for event in pygame.event.get():
        if event.type == QUIT:
            pygame.quit()
            sys.exit()
        if event.type == KEYDOWN:
            if event.key == K_ESCAPE:
                running = False
    pygame.display.update()
    mainClock.tick(60)
ta()

for event in pygame.event.get():
    if event.type == QUIT:
        pygame.quit()
        sys.exit()
    if event.type == KEYDOWN:
        if event.key == K_ESCAPE:
            running = False
pygame.display.update()
mainClock.tick(60)

def keluar():
    running = True
    while running:
        win.fill((0,0,0))
        draw_text('Keluar', font, (255, 255, 255), win, 20, 20)
        for event in pygame.event.get():
            if event.type == QUIT:
                pygame.quit()
                sys.exit()
            if event.type == KEYDOWN:
                if event.key == K_ESCAPE:
                    running = False
        pygame.display.update()
        mainClock.tick(60)
main_menu()

```

Game

```

import pygame, sys
import os
import time
from pygame.locals import *
pygame.init()
win = pygame.display.set_mode((1024,600))
pygame.display.set_caption("Game")

walkRight = [pygame.image.load(os.path.join('jalan', 'orangjalan01.png')),
pygame.image.load(os.path.join('jalan', 'orangjalan1.png')), pygame.image.load(os.path.join('jalan',
'orangjalan2.png')), pygame.image.load(os.path.join('jalan', 'orangjalan3.png')),
pygame.image.load(os.path.join('jalan', 'orangjalan4.png')), pygame.image.load(os.path.join('jalan',

```

```

'orangjalan5.png')), pygame.image.load(os.path.join('jalan', 'orangjalan6.png')),
pygame.image.load(os.path.join('jalan', 'orangjalan7.png'))]
walkLeft = [pygame.image.load(os.path.join('jalan', 'orangjalan8.png')),
pygame.image.load(os.path.join('jalan', 'orangjalan9.png')), pygame.image.load(os.path.join('jalan',
'orangjalan10.png')), pygame.image.load(os.path.join('jalan', 'orangjalan11.png')),
pygame.image.load(os.path.join('jalan', 'orangjalan12.png')),
pygame.image.load(os.path.join('jalan', 'orangjalan13.png')),
pygame.image.load(os.path.join('jalan', 'orangjalan14.png'))]
bg = pygame.image.load(os.path.join('menu', 'bg1.png'))
img1 = pygame.image.load(os.path.join('alat', 'pacul.png'))
img2 = pygame.image.load(os.path.join('alat', 'jagung.png'))
img3 = pygame.image.load(os.path.join('alat', 'air.png'))
img4 = pygame.image.load(os.path.join('alat', 'pupuk.png'))
img5 = pygame.image.load(os.path.join('alat', 'pupuk2.png'))
img6 = pygame.image.load(os.path.join('alat', 'racun.png'))
img7 = pygame.image.load(os.path.join('menu', 'awal.png'))
char = pygame.image.load(os.path.join('jalan', 'orangjalan01.png'))

```

```

sounds = {"click":pygame.mixer.Sound("musik/tuk.wav")}
pygame.mixer.music.load("Musik.wav")
pygame.mixer.music.play(-1,0.0)
pygame.mixer.music.set_volume(0.25)
clock = pygame.time.Clock()
font = pygame.font.SysFont('comicsans', 30, True)

```

```

class player(object):
    def __init__(self,x,y,width,height):
        self.x = 400
        self.y = 400
        self.width = width
        self.height = height
        self.vel = 6
        self.left = False
        self.right = False
        self.walkCount = 1
        self.standing = True
        self.hitbox = (self.x, self.y, 0, 0)

    def draw(self, win):
        if self.walkCount + 1 >= 27:
            self.walkCount = 0
        if not(self.standing):
            if self.left:
                win.blit(walkLeft[self.walkCount//4], (self.x,self.y))
                self.walkCount += 1
            elif self.right:
                win.blit(walkRight[self.walkCount//4], (self.x,self.y))
                self.walkCount += 1
        else:
            if self.right:
                win.blit(char, (self.x, self.y))
            else:
                win.blit(char, (self.x, self.y))
        self.hitbox = (self.x+40, self.y+15, 30, 120)
        #pygame.draw.rect(win, (255,0,0), self.hitbox,2)

```

```

class tanah(object):
    th = pygame.image.load(os.path.join('alat', 'tanah.png'))

    def __init__(self,x,y,width,height):
        self.x = 730
        self.y = 525
        self.width = width
        self.height = height
        self.mati = True

    def draw(self, win):
        if not(self.mati):
            win.blit(self.th, (self.x,self.y))

def pacul():
    running = True
    while running:
        #kegiatan memacul
        pygame.init()
        W, H = 1024, 600
        win = pygame.display.set_mode((W,H))
        pygame.display.set_caption('Kegiatan Memacul')
        font = pygame.font.SysFont('comicans', 30, True)
        gt = pygame.image.load(os.path.join('text', 'memacul.png'))
        clock = pygame.time.Clock()

class player(object):
    run = [pygame.image.load(os.path.join('memacul', str(x) + '.png')) for x in range(1, 9)]
    def __init__(self, x, y, width, height):
        self.x = x
        self.y = y
        self.width = width
        self.height = height
        self.jumping = False
        self.sliding = False
        self.falling = False
        self.slideCount = 0
        self.jumpCount = 0
        self.runCount = 0
        self.slideUp = False

    def draw(self, win):
        if self.falling:
            win.blit(self.fall, (self.x, self.y + 30))
        elif self.jumping:
            self.y -= self.jumpList[self.jumpCount] * 1.3
            win.blit(self.jump[self.jumpCount//18], (self.x, self.y))
            self.jumpCount += 1
            if self.jumpCount > 108:
                self.jumpCount = 0
                self.jumping = False
                self.runCount = 0
                self.hitbox = (self.x+ 4, self.y, self.width-24, self.height-10)
        elif self.sliding or self.slideUp:
            if self.slideCount < 20:

```

```

        self.y += 1
        self.hitbox = (self.x+ 4, self.y, self.width-24, self.height-10)
    elif self.slideCount == 80:
        self.y -= 19
        self.sliding = False
        self.slideUp = True
    elif self.slideCount > 20 and self.slideCount < 80:
        self.hitbox = (self.x, self.y+3, self.width-8, self.height-35)
    if self.slideCount >= 110:
        self.slideCount = 0
        self.runCount = 0
        self.slideUp = False
        self.hitbox = (self.x+ 4, self.y, self.width-24, self.height-10)
    win.blit(self.slide[self.slideCount//10], (self.x, self.y))
    self.slideCount += 1
else:
    if self.runCount > 42:
        self.runCount = 0
        win.blit(self.run[self.runCount//6], (self.x,self.y))
        self.runCount += 1
        self.hitbox = (self.x+ 4, self.y, self.width-24, self.height-13)
    #pygame.draw.rect(win, (255,0,0),self.hitbox, 2)

def p():
    tm = font.render('Timer:' + str(timer), 1, (0, 0, 0))
    win.blit(bg, (0, 0))
    win.blit(tm, (10, 10))
    win.blit(gt, (510, 200))
    runner.draw(win)
    pygame.display.update()

speed = 30
timer = 0
score = 0
run = True
runner = player(690, 395, 64, 64)

pygame.time.set_timer(USEREVENT + 1, 250)
while run:
    for event in pygame.event.get():
        if event.type == pygame.QUIT:
            pygame.quit()
            run = False
        if event.type == USEREVENT+1:
            speed -= 1

    timer = speed//5 - 1
    if timer < 1:
        tn.mati = False
        run = False
        running = False
        clock.tick(speed)
        p()

    for event in pygame.event.get():

```

```

        if event.type == QUIT:
            pygame.quit()
            sys.exit()
        if event.type == KEYDOWN:
            if event.key == K_ESCAPE:
                running = False
        pygame.display.update()
        clock.tick(60)

def jagung():
    running = True
    while running:
        #kegiatan menanam jagung
        pygame.init()
        W, H = 1024, 600
        win = pygame.display.set_mode((W,H))
        pygame.display.set_caption('Kegiatan Menanam')
        font = pygame.font.SysFont('comicsans', 30, True)
        th = pygame.image.load(os.path.join('alat', 'tanah.png'))
        gt = pygame.image.load(os.path.join('text', 'menanam.png'))
        clock = pygame.time.Clock()

class player(object):
    run = [pygame.image.load(os.path.join('menanam', str(x) + '.png')) for x in range(1, 9)]
    def __init__(self, x, y, width, height):
        self.x = x
        self.y = y
        self.width = width
        self.height = height
        self.jumping = False
        self.sliding = False
        self.falling = False
        self.slideCount = 0
        self.jumpCount = 0
        self.runCount = 0
        self.slideUp = False

    def draw(self, win):
        if self.falling:
            win.blit(self.fall, (self.x, self.y + 30))
        elif self.jumping:
            self.y -= self.jumpList[self.jumpCount] * 1.3
            win.blit(self.jump[self.jumpCount//18], (self.x, self.y))
            self.jumpCount += 1
            if self.jumpCount > 108:
                self.jumpCount = 0
                self.jumping = False
                self.runCount = 0
            self.hitbox = (self.x+ 4, self.y, self.width-24, self.height-10)
        elif self.sliding or self.slideUp:
            if self.slideCount < 20:
                self.y += 1
                self.hitbox = (self.x+ 4, self.y, self.width-24, self.height-10)
            elif self.slideCount == 80:
                self.y -= 19

```

```

        self.sliding = False
        self.slideUp = True
    elif self.slideCount > 20 and self.slideCount < 80:
        self.hitbox = (self.x, self.y+3, self.width-8, self.height-35)
    if self.slideCount >= 110:
        self.slideCount = 0
        self.runCount = 0
        self.slideUp = False
        self.hitbox = (self.x+ 4, self.y, self.width-24, self.height-10)
    win.blit(self.slide[self.slideCount//10], (self.x, self.y))
    self.slideCount += 1
else:
    if self.runCount > 42:
        self.runCount = 0
        win.blit(self.run[self.runCount//6], (self.x,self.y))
        self.runCount += 1
        self.hitbox = (self.x+ 4, self.y, self.width-24, self.height-13)
    #pygame.draw.rect(win, (255,0,0),self.hitbox, 2)

def q():
    tm = font.render('Timer:' + str(timer), 1, (0, 0, 0))
    win.blit(bg, (0, 0))
    win.blit(th, (730, 525))
    win.blit(gt, (510, 200))
    win.blit(tm, (10, 10))
    runner.draw(win)
    pygame.display.update()

speed = 30
timer = 0
score = 0
run = True
runner = player(670, 410, 64, 64)
pygame.time.set_timer(USEREVENT + 1, 250)

while run:
    clock.tick(speed)
    q()
    for event in pygame.event.get():
        if event.type == QUIT:
            pygame.quit()
            run = False
        if event.type == USEREVENT+1:
            speed -= 1

    timer = speed//5 - 1
    if timer < 1:
        tn.mati = False
        run = False
        global tambah
        tambah = 1
        running = False
        clock.tick(speed)
        q()

```

```

for event in pygame.event.get():
    if event.type == QUIT:
        pygame.quit()
        sys.exit()
    if event.type == KEYDOWN:
        if event.key == K_ESCAPE:
            running = False
pygame.display.update()
clock.tick(60)

def air():
    running = True
    global speed2
    speed2 = 35
    while running:
        #kegiatan menyiram tanaman
        pygame.init()
        W, H = 1024, 600
        win = pygame.display.set_mode((W,H))
        pygame.display.set_caption('Kegiatan Menyiram')
        font = pygame.font.SysFont('comicsans', 30, True)
        th = pygame.image.load(os.path.join('alat', 'tanah.png'))
        gt = pygame.image.load(os.path.join('text', 'menyiram.png'))
        jk0 = pygame.image.load(os.path.join('jagung', '0.png'))
        clock = pygame.time.Clock()

class player(object):
    run = [pygame.image.load(os.path.join('menyiram', str(x) + '.png')) for x in range(1, 9)]
    def __init__(self, x, y, width, height):
        self.x = x
        self.y = y
        self.width = width
        self.height = height
        self.jumping = False
        self.sliding = False
        self.falling = False
        self.slideCount = 0
        self.jumpCount = 0
        self.runCount = 0
        self.slideUp = False

    def draw(self, win):
        if self.falling:
            win.blit(self.fall, (self.x, self.y + 30))
        elif self.jumping:
            self.y -= self.jumpList[self.jumpCount] * 1.3
            win.blit(self.jump[self.jumpCount//18], (self.x, self.y))
            self.jumpCount += 1
            if self.jumpCount > 108:
                self.jumpCount = 0
                self.jumping = False
                self.runCount = 0
                self.hitbox = (self.x+ 4, self.y, self.width-24, self.height-10)
        elif self.sliding or self.slideUp:
            if self.slideCount < 20:

```

```

        self.y += 1
        self.hitbox = (self.x+ 4, self.y, self.width-24, self.height-10)
    elif self.slideCount == 80:
        self.y -= 19
        self.sliding = False
        self.slideUp = True
    elif self.slideCount > 20 and self.slideCount < 80:
        self.hitbox = (self.x, self.y+3, self.width-8, self.height-35)
    if self.slideCount >= 110:
        self.slideCount = 0
        self.runCount = 0
        self.slideUp = False
        self.hitbox = (self.x+ 4, self.y, self.width-24, self.height-10)
    win.blit(self.slide[self.slideCount//10], (self.x, self.y))
    self.slideCount += 1
else:
    if self.runCount > 42:
        self.runCount = 0
        win.blit(self.run[self.runCount//6], (self.x,self.y))
        self.runCount += 1
        self.hitbox = (self.x+ 4, self.y, self.width-24, self.height-13)
    #pygame.draw.rect(win, (255,0,0),self.hitbox, 2)

def r():
    tm = font.render('Timer:' + str(timer), 1, (0, 0, 0))
    win.blit(bg, (0, 0))
    win.blit(th, (730, 525))
    win.blit(gt, (510, 200))
    win.blit(jk0, (746, 525))
    win.blit(tm, (10, 10))
    runner.draw(win)
    pygame.display.update()

speed = 30
timer = 0
score = 0
run = True
runner = player(670, 410, 64, 64)
pygame.time.set_timer(USEREVENT + 1, 400)

while run:
    for event in pygame.event.get():
        if event.type == pygame.QUIT:
            pygame.quit()
            run = False
        if event.type == USEREVENT+1:
            speed -= 1

    timer = speed//5 - 1
    if timer < 1:
        tn.mati = False
        run = False
        end.run = False
        running = False
        clock.tick(speed)

```

```

r()

for event in pygame.event.get():
    if event.type == QUIT:
        pygame.quit()
        sys.exit()
    if event.type == KEYDOWN:
        if event.key == K_ESCAPE:
            running = False
pygame.display.update()
clock.tick(60)

def pupuk():
    running = True
    global speed2
    speed2 = 35
    while running:
        #kegiatan memupuk
        pygame.init()
        W, H = 1024, 600
        win = pygame.display.set_mode((W,H))
        pygame.display.set_caption('Kegiatan Memupuk')
        font = pygame.font.SysFont('comicans', 30, True)
        th = pygame.image.load(os.path.join('alat', 'tanah.png'))
        gt = pygame.image.load(os.path.join('text', 'memupuk.png'))
        jkl = pygame.image.load(os.path.join('jagung', '1.png'))
        clock = pygame.time.Clock()

class player(object):
    run = [pygame.image.load(os.path.join('memupuk', str(x) + '.png')) for x in range(1, 9)]

    def __init__(self, x, y, width, height):
        self.x = x
        self.y = y
        self.width = width
        self.height = height
        self.jumping = False
        self.sliding = False
        self.falling = False
        self.slideCount = 0
        self.jumpCount = 0
        self.runCount = 0
        self.slideUp = False

    def draw(self, win):
        if self.falling:
            win.blit(self.fall, (self.x, self.y + 30))
        elif self.jumping:
            self.y -= self.jumpList[self.jumpCount] * 1.3
            win.blit(self.jump[self.jumpCount//18], (self.x, self.y))
            self.jumpCount += 1
            if self.jumpCount > 108:
                self.jumpCount = 0
                self.jumping = False
                self.runCount = 0

```

```

        self.hitbox = (self.x+ 4, self.y, self.width-24, self.height-10)
    elif self.sliding or self.slideUp:
        if self.slideCount < 20:
            self.y += 1
            self.hitbox = (self.x+ 4, self.y, self.width-24, self.height-10)
        elif self.slideCount == 80:
            self.y -= 19
            self.sliding = False
            self.slideUp = True
        elif self.slideCount > 20 and self.slideCount < 80:
            self.hitbox = (self.x, self.y+3, self.width-8, self.height-35)
        if self.slideCount >= 110:
            self.slideCount = 0
            self.runCount = 0
            self.slideUp = False
            self.hitbox = (self.x+ 4, self.y, self.width-24, self.height-10)
        win.blit(self.slide[self.slideCount//10], (self.x, self.y))
        self.slideCount += 1

    else:
        if self.runCount > 42:
            self.runCount = 0
            win.blit(self.run[self.runCount//6], (self.x,self.y))
            self.runCount += 1
            self.hitbox = (self.x+ 4, self.y, self.width-24, self.height-13)
        #pygame.draw.rect(win, (255,0,0),self.hitbox, 2)

def s():
    tm = font.render("Timer:' + str(timer), 1, (0, 0, 0))
    win.blit(bg, (0, 0))
    win.blit(th, (730, 525))
    win.blit(gt, (510, 200))
    win.blit(jk1, (721, 498))
    win.blit(tm, (10, 10))
    runner.draw(win)
    pygame.display.update()

speed = 30
timer = 0
score = 0
run = True
runner = player(660, 410, 64, 64)
pygame.time.set_timer(USEREVENT + 1, 450)

while run:
    clock.tick(speed)
    s()

    for event in pygame.event.get():
        if event.type == QUIT:
            pygame.quit()
            run = False
        if event.type == USEREVENT+1:
            speed -= 1

```

```

timer = speed//5 - 1
if timer < 1:
    tn.mati = False
    run = False
    running = False
    clock.tick(speed)
    s()

for event in pygame.event.get():
    if event.type == QUIT:
        pygame.quit()
        sys.exit()
    if event.type == KEYDOWN:
        if event.key == K_ESCAPE:
            running = False
pygame.display.update()
clock.tick(60)

def pupuk2():
    running = True
    while running:
        #kegiatan penyiangan
        pygame.init()
        W, H = 1024, 600
        win = pygame.display.set_mode((W,H))
        pygame.display.set_caption('Kegiatan Menyiram')
        font = pygame.font.SysFont('comicsans', 30, True)
        th = pygame.image.load(os.path.join('alat', 'tanah.png'))
        gt = pygame.image.load(os.path.join('text', 'menyiang.png'))
        jk2 = pygame.image.load(os.path.join('jagung', '2.png'))
        clock = pygame.time.Clock()

class player(object):
    run = [pygame.image.load(os.path.join('menyemprot', str(x) + '.png')) for x in range(1, 9)]

    def __init__(self, x, y, width, height):
        self.x = x
        self.y = y
        self.width = width
        self.height = height
        self.jumping = False
        self.sliding = False
        self.falling = False
        self.slideCount = 0
        self.jumpCount = 0
        self.runCount = 0
        self.slideUp = False

    def draw(self, win):
        if self.falling:
            win.blit(self.fall, (self.x, self.y + 30))
        elif self.jumping:
            self.y -= self.jumpList[self.jumpCount] * 1.3
            win.blit(self.jump[self.jumpCount//18], (self.x, self.y))

```

```

        self.jumpCount += 1
        if self.jumpCount > 108:
            self.jumpCount = 0
            self.jumping = False
            self.runCount = 0
            self.hitbox = (self.x+ 4, self.y, self.width-24, self.height-10)
        elif self.sliding or self.slideUp:
            if self.slideCount < 20:
                self.y += 1
                self.hitbox = (self.x+ 4, self.y, self.width-24, self.height-10)
            elif self.slideCount == 80:
                self.y -= 19
                self.sliding = False
                self.slideUp = True
            elif self.slideCount > 20 and self.slideCount < 80:
                self.hitbox = (self.x, self.y+3, self.width-8, self.height-35)
            if self.slideCount >= 110:
                self.slideCount = 0
                self.runCount = 0
                self.slideUp = False
                self.hitbox = (self.x+ 4, self.y, self.width-24, self.height-10)
            win.blit(self.slide[self.slideCount//10], (self.x, self.y))
            self.slideCount += 1
        else:
            if self.runCount > 42:
                self.runCount = 0
                win.blit(self.run[self.runCount//6], (self.x,self.y))
                self.runCount += 1
                self.hitbox = (self.x+ 4, self.y, self.width-24, self.height-13)
            #pygame.draw.rect(win, (255,0,0),self.hitbox, 2)

def t():
    tm = font.render('Timer:' + str(timer), 1, (0, 0, 0))
    win.blit(bg, (0, 0))
    win.blit(th, (730, 525))
    win.blit(gt, (510, 200))
    win.blit(jk2, (722, 499))
    win.blit(tm, (10, 10))
    runner.draw(win)
    pygame.display.update()

speed = 30
timer = 0
score = 0
run = True
runner = player(640, 371, 64, 64)
pygame.time.set_timer(USEREVENT + 1, 500)

while run:
    clock.tick(speed)
    t()

    for event in pygame.event.get():
        if event.type == QUIT:
            pygame.quit()

```

```

        run = False
        if event.type == USEREVENT+1:
            speed -= 1

    timer = speed//5 - 1
    if timer < 1:
        tn.mati = False
        run = False
        running = False
        clock.tick(speed)
        t()

for event in pygame.event.get():
    if event.type == QUIT:
        pygame.quit()
        sys.exit()
    if event.type == KEYDOWN:
        if event.key == K_ESCAPE:
            running = False
pygame.display.update()
clock.tick(60)

def racun():
    global speed2
    speed2 = 35
    running = True
    while running:
        #kegiatan basmi hama tanaman
        pygame.init()
        W, H = 1024, 600
        win = pygame.display.set_mode((W,H))
        pygame.display.set_caption('Kegiatan Menyiram')
        font = pygame.font.SysFont('comicsans', 30, True)
        th = pygame.image.load(os.path.join('alat', 'tanah.png'))
        gt = pygame.image.load(os.path.join('text', 'menyemprot.png'))
        jk2 = pygame.image.load(os.path.join('jagung', '2.png'))
        clock = pygame.time.Clock()

class player(object):
    run = [pygame.image.load(os.path.join('menyemprot', str(x) + '.png')) for x in range(1, 9)]

    def __init__(self, x, y, width, height):
        self.x = x
        self.y = y
        self.width = width
        self.height = height
        self.jumping = False
        self.sliding = False
        self.falling = False
        self.slideCount = 0
        self.jumpCount = 0
        self.runCount = 0
        self.slideUp = False

    def draw(self, win):
        if self.falling:

```

```

        win.blit(self.fall, (self.x, self.y + 30))
    elif self.jumping:
        self.y -= self.jumpList[self.jumpCount] * 1.3
        win.blit(self.jump[self.jumpCount//18], (self.x, self.y))
        self.jumpCount += 1
    if self.jumpCount > 108:
        self.jumpCount = 0
        self.jumping = False
        self.runCount = 0
        self.hitbox = (self.x+ 4, self.y, self.width-24, self.height-10)
    elif self.sliding or self.slideUp:
        if self.slideCount < 20:
            self.y += 1
            self.hitbox = (self.x+ 4, self.y, self.width-24, self.height-10)
        elif self.slideCount == 80:
            self.y -= 19
            self.sliding = False
            self.slideUp = True
        elif self.slideCount > 20 and self.slideCount < 80:
            self.hitbox = (self.x, self.y+3, self.width-8, self.height-35)
        if self.slideCount >= 110:
            self.slideCount = 0
            self.runCount = 0
            self.slideUp = False
            self.hitbox = (self.x+ 4, self.y, self.width-24, self.height-10)
        win.blit(self.slide[self.slideCount//10], (self.x, self.y))
        self.slideCount += 1
    else:
        if self.runCount > 42:
            self.runCount = 0
            win.blit(self.run[self.runCount//6], (self.x,self.y))
            self.runCount += 1
            self.hitbox = (self.x+ 4, self.y, self.width-24, self.height-13)

#pygame.draw.rect(win, (255,0,0),self.hitbox, 2)

def u():
    tm = font.render('Timer:' + str(timer), 1, (0, 0, 0))
    win.blit(bg, (0, 0))
    win.blit(th, (730, 525))
    win.blit(gt, (510, 200))
    win.blit(jk1, (721, 498))
    win.blit(tm, (10, 10))
    runner.draw(win)
    pygame.display.update()

speed = 30
timer = 0
score = 0
run = True
runner = player(640, 371, 64, 64)
pygame.time.set_timer(USEREVENT + 1, 500)

while run:
    clock.tick(speed)

```

```

u()

for event in pygame.event.get():
    if event.type == QUIT:
        pygame.quit()
        run = False
    if event.type == USEREVENT+1:
        speed -= 1

timer = speed//5 - 1
if timer < 1:
    tn.mati = False
    run = False
    running = False
    clock.tick(speed)
    u()

for event in pygame.event.get():
    if event.type == QUIT:
        pygame.quit()
        sys.exit()
    if event.type == KEYDOWN:
        if event.key == K_ESCAPE:
            running = False
pygame.display.update()
clock.tick(60)

def end():
    global speed1,speed2, timer1, timer2, tambah, tn
    speed = 0
    speed1 = 0
    timer1 = 0
    timer2 = 0
    speed2 = 35
    tambah = 0
    man = player(200, 410, 64, 64)
    tn = tanah(0, 0, 0, 0)
    ulang = pygame.image.load(os.path.join('gagal', 'Y.png'))
    menu = pygame.image.load(os.path.join('gagal', 'T.png'))
    img3 = pygame.image.load(os.path.join('gagal', 'tex.png'))

    pygame.mixer.music.load("lose.wav")
    pygame.mixer.music.play(0,0.0)
    pygame.mixer.music.set_volume(0.25)

run = True
while run:
    pygame.time.delay(1000)
    click = False
    for event in pygame.event.get():
        if event.type == pygame.QUIT:
            run = False
            pygame.quit()
        if event.type == pygame.MOUSEBUTTONDOWN:
            if event.button == 1:

```

```

        click = True

    mx, my = pygame.mouse.get_pos()
    button_1 = pygame.Rect(340, 350, 230, 50)
    button_2 = pygame.Rect(570, 350, 230, 50)
    button_3 = pygame.Rect(270, 170, 230, 50)
    if button_1.collidepoint((mx, my)):
        if click:
            sounds["click"].play()
            pygame.time.delay(500)
            pygame.mixer.music.load("Musik.wav")
            pygame.mixer.music.play(0,0.0)
            pygame.mixer.music.set_volume(0.25)
            run = False

    if button_2.collidepoint((mx, my)):
        if click:
            sounds["click"].play()
            pygame.time.delay(500)
            pygame.quit()
    win.blit(bg, (0,0))
    win.blit(ulang, button_1)
    win.blit(menu, button_2)
    win.blit(img3, button_3)
    #pygame.draw.rect(win, (255, 0, 0), button_1)
    #pygame.draw.rect(win, (255, 0, 0), button_2)
    #pygame.draw.rect(win, (255, 0, 0), button_3)
    pygame.display.update()

def game():
    running = True
    while running:
        import game
        for event in pygame.event.get():
            if event.type == QUIT:
                pygame.quit()
                sys.exit()
            if event.type == KEYDOWN:
                if event.key == K_ESCAPE:
                    running = False
        pygame.display.update()
        clock.tick(60)

def mn():
    running = True
    while running:
        import menu_utama
        menu_utama.main_menu()
        for event in pygame.event.get():
            if event.type == QUIT:
                pygame.quit()
                sys.exit()
            if event.type == KEYDOWN:
                if event.key == K_ESCAPE:
                    running = False

```

```

pygame.display.update()
mainClock.tick(60)

def sukses():
    running = True
    while running:
        pygame.display.set_caption('game')
        mainClock = pygame.time.Clock()
        win = pygame.display.set_mode((1024, 600), 0, 32)
        bg = pygame.image.load(os.path.join('gagal', 'bg1.png'))
        font = pygame.font.SysFont(None, 20)

        sounds = {'click': pygame.mixer.Sound("Musik/click.wav")}
        pygame.mixer.music.load("win.wav")
        pygame.mixer.music.play(0, 0.0)
        pygame.mixer.music.set_volume(0.25)

    def at():
        while True:
            win.blit(bg, (0, 0))
            img1 = pygame.image.load(os.path.join('menu', 'awal.png'))
            mx, my = pygame.mouse.get_pos()
            button_1 = pygame.Rect(20, 535, 70, 60)

            if button_1.collidepoint((mx, my)):
                if click:
                    sounds["click"].play()
                    game()
            #pygame.draw.rect(win, (255, 0, 0), button_1)
            win.blit(img1, button_1)

            click = False
            for event in pygame.event.get():
                if event.type == QUIT:
                    pygame.quit()
                    sys.exit()
                if event.type == KEYDOWN:
                    if event.key == K_ESCAPE:
                        pygame.quit()
                        sys.exit()
                if event.type == MOUSEBUTTONDOWN:
                    if event.button == 1:
                        pygame.time.delay(500)
                        click = True
            pygame.display.update()
            mainClock.tick(60)

    def game():
        running = True
        while running:
            import menu_utama
            for event in pygame.event.get():
                if event.type == QUIT:
                    pygame.quit()
                    sys.exit()

```

```

        if event.type == KEYDOWN:
            if event.key == K_ESCAPE:
                running = False
            pygame.display.update()
            mainClock.tick(60)
    at()

    for event in pygame.event.get():
        if event.type == QUIT:
            pygame.quit()
            sys.exit()
        if event.type == KEYDOWN:
            if event.key == K_ESCAPE:
                running = False
            pygame.display.update()
            mainClock.tick(60)

def redrawGameWindow():
    tm = font.render('Hari:' + str(timer1), 1, (0, 0, 0))
    tm1 = font.render('Error:' + str(timer2), 1, (0, 0, 0))
    win.blit(bg, (0,0))
    win.blit(tm, (10, 10))
    win.blit(tm1, (500, 10))
    man.draw(win)
    tn.draw(win)
    #pygame.draw.rect(win, (255, 0, 0), button_1)
    win.blit(img1, button_1)
    #pygame.draw.rect(win, (255, 0, 0), button_2)
    win.blit(img2, button_2)
    #pygame.draw.rect(win, (255, 0, 0), button_3)
    win.blit(img3, button_3)
    #pygame.draw.rect(win, (255, 0, 0), button_4)
    win.blit(img4, button_4)
    #pygame.draw.rect(win, (255, 0, 0), button_5)
    win.blit(img5, button_5)
    #pygame.draw.rect(win, (255, 0, 0), button_6)
    win.blit(img6, button_6)
    #pygame.draw.rect(win, (255, 0, 0), button_7)
    win.blit(img7, button_7)
    pygame.display.update()

man = player(200, 410, 64, 64)
tn = tanah(10, 10, 64, 64)
tx1 = pygame.image.load(os.path.join('text', '1.png'))
tx2 = pygame.image.load(os.path.join('text', '2.png'))
tx3 = pygame.image.load(os.path.join('text', '3.png'))
tx4 = pygame.image.load(os.path.join('text', '4.png'))
tx5 = pygame.image.load(os.path.join('text', '5.png'))
tx6 = pygame.image.load(os.path.join('text', '6.png'))
tx7 = pygame.image.load(os.path.join('text', '7.png'))
tx8 = pygame.image.load(os.path.join('text', '8.png'))
tx9 = pygame.image.load(os.path.join('text', '9.png'))
tx10 = pygame.image.load(os.path.join('text', '10.png'))
tx11 = pygame.image.load(os.path.join('text', '11.png'))
tx12 = pygame.image.load(os.path.join('text', '12.png'))

```

```

tx13 = pygame.image.load(os.path.join('text', '13.png'))
tx14 = pygame.image.load(os.path.join('text', '14.png'))

hml = pygame.image.load(os.path.join('hama', '1.png'))
ht2 = pygame.image.load(os.path.join('text', 'ht2.png'))

jk0 = pygame.image.load(os.path.join('jagung', '0.png'))
jk1 = pygame.image.load(os.path.join('jagung', '1.png'))
jk2 = pygame.image.load(os.path.join('jagung', '2.png'))
jk3 = pygame.image.load(os.path.join('jagung', '3.png'))
jk4 = pygame.image.load(os.path.join('jagung', '4.png'))
jk5 = pygame.image.load(os.path.join('jagung', '5.png'))
jk6 = pygame.image.load(os.path.join('jagung', '6.png'))
jk7 = pygame.image.load(os.path.join('jagung', '7.png'))
jk8 = pygame.image.load(os.path.join('jagung', '8.png'))
jk9 = pygame.image.load(os.path.join('jagung', '9.png'))
jk10 = pygame.image.load(os.path.join('jagung', '10.png'))
jk11 = pygame.image.load(os.path.join('jagung', '11.png'))
jk12 = pygame.image.load(os.path.join('jagung', '12.png'))
jk13 = pygame.image.load(os.path.join('jagung', '13.png'))

tambah = 0
timer1 = 0
timer2 = 0
speed1 = 0
speed2 = 35
pygame.time.set_timer(USEREVENT + 1, 50)
#mainloop
run = True
while run:
    clock.tick(27)
    if timer1 == 3:
        win.blit(jk0, (746, 525))
    if timer1 == 4:
        win.blit(jk0, (746, 525))
    if timer1 == 4:
        win.blit(tx2, (500, 200))
    if timer1 == 5:
        win.blit(jk0, (746, 525))
    if timer1 == 6:
        win.blit(jk0, (746, 525))
    if timer2 == 1:
        end()
    if timer1 == 7:
        win.blit(tx3, (490, 200))
    if timer1 == 7:
        win.blit(jk0, (746, 525))
    if timer1 == 8:
        win.blit(jk1, (721, 498))
    if timer1 == 9:
        win.blit(jk1, (721, 498))
    if timer1 == 9:
        win.blit(tx5, (500, 200))
    if timer1 == 10:
        win.blit(jk1, (721, 498))

```

```

if timer1 == 11:
    win.blit(jk1, (721, 498))
if timer1 == 12:
    win.blit(jk1, (721, 498))
if timer2 == 1:
    end()
if timer1 == 13:
    win.blit(jk1, (721, 498))
if timer1 == 13:
    win.blit(tx6, (450, 200))
if timer1 == 14:
    win.blit(jk2, (722, 499))
if timer1 == 14:
    win.blit(tx8, (500, 200))
if timer1 == 15:
    win.blit(jk2, (722, 499))
if timer1 == 16:
    win.blit(jk2, (722, 499))
if timer1 == 16:
    win.blit(ht2, (490, 200))
if timer1 == 16:
    win.blit(hm1, (740, 320))
if timer1 == 17:
    win.blit(jk2, (722, 499))
if timer1 == 18:
    win.blit(jk2, (722, 499))
if timer2 == 1:
    end()
if timer1 == 19:
    win.blit(jk2, (722, 499))
if timer1 == 20:
    win.blit(jk2, (722, 499))
if timer1 == 19:
    win.blit(tx9, (470, 200))
if timer1 == 21:
    win.blit(tx7, (500, 200))
if timer1 == 21:
    win.blit(jk3, (727, 476))
if timer1 == 22:
    win.blit(jk3, (727, 476))
if timer1 == 23:
    win.blit(jk3, (727, 476))
if timer1 == 24:
    win.blit(jk3, (727, 476))
if timer1 == 25:
    win.blit(jk3, (727, 476))
if timer1 == 26:
    win.blit(jk3, (727, 476))
if timer1 == 27:
    win.blit(jk3, (727, 476))
if timer1 == 28:
    win.blit(tx10, (500, 200))
if timer1 == 28:
    win.blit(jk4, (728, 455))
if timer1 == 29:

```

```
    win.blit(jk4, (728, 455))
if timer1 == 30:
    win.blit(jk4, (728, 455))
if timer1 == 31:
    win.blit(jk4, (728, 455))
if timer1 == 32:
    win.blit(jk4, (728, 455))
if timer1 == 33:
    win.blit(jk4, (728, 455))
if timer1 == 34:
    win.blit(jk4, (728, 455))
if timer1 == 35:
    win.blit(tx11, (500, 200))
if timer1 == 35:
    win.blit(jk5, (726, 389))
if timer1 == 36:
    win.blit(jk5, (726, 389))
if timer1 == 37:
    win.blit(jk5, (726, 389))
if timer1 == 38:
    win.blit(jk5, (726, 389))
if timer1 == 39:
    win.blit(jk5, (726, 389))
if timer1 == 40:
    win.blit(jk5, (726, 389))
if timer1 == 41:
    win.blit(jk5, (726, 389))
if timer1 == 42:
    win.blit(tx12, (500, 200))
if timer1 == 42:
    win.blit(jk6, (727, 411))
if timer1 == 43:
    win.blit(jk6, (727, 411))
if timer1 == 44:
    win.blit(jk6, (727, 411))
if timer1 == 45:
    win.blit(jk6, (727, 411))
if timer1 == 46:
    win.blit(jk6, (727, 411))
if timer1 == 47:
    win.blit(jk6, (727, 411))
if timer1 == 48:
    win.blit(jk6, (727, 411))
if timer1 == 49:
    win.blit(tx13, (500, 200))
if timer1 == 49:
    win.blit(jk7, (726, 375))
if timer1 == 50:
    win.blit(jk7, (726, 375))
if timer1 == 51:
    win.blit(jk7, (726, 375))
if timer1 == 52:
    win.blit(jk7, (726, 375))
if timer1 == 53:
    win.blit(jk7, (726, 375))
```

```
if timer1 == 54:
    win.blit(jk7, (726, 375))
if timer1 == 55:
    win.blit(jk7, (726, 375))
if timer1 == 56:
    win.blit(tx14, (500, 200))
if timer1 == 56:
    win.blit(jk8, (726, 350))
if timer1 == 57:
    win.blit(jk8, (726, 350))
if timer1 == 58:
    win.blit(jk8, (726, 350))
if timer1 == 59:
    win.blit(jk8, (726, 350))
if timer1 == 60:
    win.blit(jk8, (726, 350))
if timer1 == 61:
    win.blit(jk8, (726, 350))
if timer1 == 62:
    win.blit(jk8, (726, 350))
if timer1 == 63:
    win.blit(jk10, (726, 350))
if timer1 == 64:
    win.blit(jk10, (726, 350))
if timer1 == 65:
    win.blit(jk10, (726, 350))
if timer1 == 66:
    win.blit(jk10, (726, 350))
if timer1 == 67:
    win.blit(jk10, (726, 350))
if timer1 == 68:
    win.blit(jk10, (726, 350))
if timer1 == 69:
    win.blit(jk1, (726, 350))
if timer1 == 70:
    win.blit(jk11, (726, 350))
if timer1 == 71:
    win.blit(jk11, (726, 350))
if timer1 == 72:
    win.blit(jk11, (726, 350))
if timer1 == 73:
    win.blit(jk11, (726, 350))
if timer1 == 74:
    win.blit(jk12, (726, 350))
if timer1 == 75:
    win.blit(jk12, (726, 350))
if timer1 == 76:
    win.blit(jk12, (726, 350))
if timer1 == 77:
    win.blit(jk12, (726, 350))
if timer1 == 78:
    win.blit(jk12, (726, 350))
if timer1 == 79:
    win.blit(jk13, (726, 350))
if timer1 == 80:
```

```

        win.blit(jk13, (726, 350))
    if timer1 == 81:
        sukses()

    mx, my = pygame.mouse.get_pos()
    button_1 = pygame.Rect(945, 10, 70, 47)
    button_2 = pygame.Rect(945, 65, 70, 47)
    button_3 = pygame.Rect(945, 120, 70, 47)
    button_4 = pygame.Rect(945, 175, 70, 47)
    button_5 = pygame.Rect(945, 230, 70, 47)
    button_6 = pygame.Rect(945, 285, 70, 47)
    button_7 = pygame.Rect(25, 535, 55, 47)

    if button_1.collidepoint((mx, my)):
        if click:
            sounds["click"].play()
            pacul()
    if button_2.collidepoint((mx, my)):
        if click:
            sounds["click"].play()
            jagung()
    if button_3.collidepoint((mx, my)):
        if click:
            sounds["click"].play()
            air()
    if button_4.collidepoint((mx, my)):
        if click:
            sounds["click"].play()
            pupuk()
    if button_5.collidepoint((mx, my)):
        if click:
            sounds["click"].play()
            pupuk2()
    if button_6.collidepoint((mx, my)):
        if click:
            sounds["click"].play()
            racun()
    if button_7.collidepoint((mx, my)):
        if click:
            sounds["click"].play()
            pygame.mixer.music.load("musik1.wav")
            mn()

    click = False
    for event in pygame.event.get():
        if event.type == QUIT:
            pygame.quit()
            sys.exit()
        if event.type == KEYDOWN:
            if event.key == K_ESCAPE:
                pygame.quit()
                sys.exit()
        if event.type == MOUSEBUTTONDOWN:
            if event.button == 1:
                pygame.time.delay(500)

```

```

        click = True
pygame.display.update()
clock.tick(60)

for event in pygame.event.get():
    if event.type == QUIT:
        pygame.quit()
        run = False
    if tambah == 1:
        if event.type == USEREVENT+1:
            speed1 += 1
            speed2 -= 1
for event in pygame.event.get():
    if event.type == pygame.QUIT:
        run = False

keys = pygame.key.get_pressed()
if keys[pygame.K_LEFT] and man.x > man.vel:
    man.x -= man.vel
    man.left = True
    man.right = False
    man.standing = False
elif keys[pygame.K_RIGHT]:
    man.x += man.vel
    man.right = True
    man.left = False
    man.standing = False
else:
    man.right = False
    man.left = False
    man.standing = True
    man.walkCount = 0

timer1 = speed1//5
timer2 = speed2//2
redrawGameWindow()
pygame.quit()

```



KEMENTERIAN RISET, TEKNOLOGI DAN PENDIDIKAN TINGGI
LEMBAGA PENELITIAN (LEMLIT)
UNIVERSITAS ICHSAN GORONTALO

Jl. Raden Saleh No. 17 Kota Gorontalo
Telp: (0435) 8724466, 829975; Fax: (0435) 82997;
E-mail: lembagapenelitian@unisan.ac.id

Nomor : 1824/PIP/LEMLIT-UNISAN/GTO/XI/2019

Lampiran : -

Hal : Permohonan Izin Penelitian

Kepada Yth,

Kepala SMK Negeri 1 Wonosari

di,-

Kab. Boalemo

Yang bertanda tangan di bawah ini :

Nama : Dr. Rahmisyari, ST., SE
NIDN : 0929117202
Jabatan : Ketua Lembaga Penelitian

Meminta kesediannya untuk memberikan izin pengambilan data dalam rangka penyusunan **Proposal / Skripsi**, kepada :

Nama Mahasiswa : Rizki Malahedi
NIM : T3116154
Fakultas : Fakultas Ilmu Komputer
Program Studi : Teknik Informatika
Lokasi Penelitian : SMK Negeri 1 Wonosari
Judul Penelitian : GAME SIMULASI PEMBELAJARAN BUDIDAYA
TANAMAN JAGUNG BERBASIS REAL TIME STRATEGY

Atas kebijakan dan kerja samanya diucapkan banyak terima kasih.

Gorontalo, 15 November 2019
Kema,

Dr. Rahmisyari, ST., SE
NIDN 0929117202



PEMERINTAH PROVINSI GORONTALO
DINAS PENDIDIKAN, KEBUDAYAAN, PEMUDA DAN OLAHRAGA
SEKOLAH MENENGAH KEJURUAN
SMK NEGERI 1 WONOSARI
Jl. Brawijaya Desa Harapan Kec. Wonosari KP.96262
Email:info@smkn1won.sch.id
KABUPATEN BOALEMO



SURAT KETERANGAN

NOMOR : 0.30../SMKN 1 WON-Kur/XI/2020

Yang bertanda tangan dibawah ini, Kepala Sekolah Menengah Kejuruan Negeri 1 Wonosari Kecamatan Wonosari Kabupaten Boalemo, menerangkan kepada :

Nama : RIZKI MALAHEDI
NIM : T3116154
Program Studi : Teknik Informartika
Fakultas : Ilmu Komputer
Alamat : Desa Dimito Kec. Wonosari Kab. Boalemo

Bahwa yang bersangkutan benar – benar mengambil lokasi penelitian di SMK Negeri 1 Wonosari berdasarkan Permohonan Izin Penelitian dari Ketua Lembaga Penelitian Universitas Ichsan Gorontalo, Nomor : 1824/PIP/LEMLIT-UNISAN/GTO/X/2020, tanggal 2 November 2020 dalam rangka penyusunan Skripsi dengan judul GAME SIMULASI PEMBELAJARAN BUDIDAYA TANAMAN JAGUNG BERBASIS REAL TIME STRATEGY.

Demikian Surat Keterangan ini dibuat untuk dipergunakan seperlunya.

Wonosari, 2 November 2020

Kepala Sekolah,


DEDE WANGSA, SST.Pi
NIP. 197907182010011004



KEMENTRIAN PENDIDIKAN DAN KEBUDAYAAN
UNIVERSITAS ICHSAN GORONTALO
UPT. PERPUSTAKAAN PUSAT

SURAT KEPUTUSAN MENDIKS RI NO. 84/D/0/2001

Jln. Achmad Nadjamuddin No.17 Telp. (0435) 829975 Fax. (0435) 829976 Gorontalo

SURAT KETERANGAN BEBAS PUSTAKA

No.003/perpus_fikom/VIII/2020

Perpustakaan Rabu, 12 Agustus 2020 Fakultas Ilmu komputer (FIKOM) Universitas Ichsan Gorontalo dengan ini menerangkan bahwa :

Nama : Rizki Malahedi
Nim : T3116154
No anggota : M201996

Terhitung sejak tanggal 12 Agustus 2020, dinyatakan telah bebas dari pinjaman buku dan koleksi lainnya diperpustakaan Fakultas Ilmu komputer.

Demikian keterangan ini di buat untuk di gunakan sebagaimana mestinya.

Gorontalo, 12 Agustus 2020
Kepala Perpustakaan
Fakultas Ilmu Komputer

Apriyanto Alhamad, M.Kom
NIDN 09240486



**KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN
UNIVERSITAS ICHSAN
(UNISAN) GORONTALO**

SURAT KEPUTUSAN MENDIKNAS RI NOMOR 84/D/O/2001

Jl. Achmad Nadjamuddin No. 17 Telp (0435) 829975 Fax (0435) 829976 Gorontalo

SURAT REKOMENDASI BEBAS PLAGIASI

No. 0881/UNISAN-G/S-BP/XII/2020

Yang bertanda tangan di bawah ini:

Nama : Sunarto Taliki, M.Kom
NIDN : 0906058301
Unit Kerja : Pustikom, Universitas Ichsan Gorontalo

Dengan ini Menyatakan bahwa :

Nama Mahasiswa : RIZKI MALAHEDI
NIM : T3118154
Program Studi : Teknik Informatika (S1)
Fakultas : Fakultas Ilmu Komputer
Judul Skripsi : Game Simulasi Pembelajaran Budidaya Tanaman Jagung Berbasis Real Time Strategy

Sesuai dengan hasil pengecekan tingkat kemiripan skripsi melalui aplikasi Turnitin untuk judul skripsi di atas diperoleh hasil Similarity sebesar 9%, berdasarkan SK Rektor No. 237/UNISAN-G/SK/IX/2019 tentang Panduan Pencegahan dan Penanggulangan Plagiarisme, bahwa batas kemiripan skripsi maksimal 35% dan sesuai dengan Surat Pernyataan dari kedua Pembimbing yang bersangkutan menyatakan bahwa isi softcopy skripsi yang diolah di Turnitin SAMA ISINYA dengan Skripsi Aslinya serta format penulisannya sudah sesuai dengan Buku Panduan Penulisan Skripsi, untuk itu skripsi tersebut di atas dinyatakan BEBAS PLAGIASI dan layak untuk diujikan.

Demikian surat rekomendasi ini dibuat untuk digunakan sebagaimana mestinya.

Gorontalo, 03 Desember 2020

Tim Verifikasi,



Sunarto Taliki, M.Kom

NIDN. 0906058301

Tembusan :

1. Dekan
2. Ketua Program Studi
3. Pembimbing I dan Pembimbing II
4. Yang bersangkutan
5. Arsip



SKRIPSI_1_T3116154_RIZKI MALAHEDI.docx

Dec 3, 2020

5742 words / 44213 characters

T3116154 RIZKI MALAHEDI

GAME SIMULASI PEMBELAJARAN BUDIDAYA TANAMAN JAG...

Sources Overview

9 %

OVERALL SIMILARITY

1	ejournal.unsrat.ac.id	INTERNET	2%
2	www.scribd.com	INTERNET	1%
3	widuri.raharja.info	INTERNET	1%
4	baper.if.uinsgd.ac.id	INTERNET	<1%
5	repository.widyatama.ac.id	INTERNET	<1%
6	journal-isi.org	INTERNET	<1%
7	jurnal.untan.ac.id	INTERNET	<1%
8	jurnal.untirta.ac.id	INTERNET	<1%
9	media.neliti.com	INTERNET	<1%
10	docplayer.info	INTERNET	<1%
11	publikasi.dinus.ac.id	INTERNET	<1%
12	www.coursehero.com	INTERNET	<1%
13	sulbar.litbang.pertanian.go.id	INTERNET	<1%

Excluded search repositories:

- Submitted Works

Excluded from Similarity Report:

- Small Matches (less than 25 words).

Excluded sources:

- None



**KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN
UNIVERSITAS ICHSAN GORONTALO
FAKULTAS ILMU KOMPUTER**

SK MENDIKNAS NOMOR 84/D/O/2001

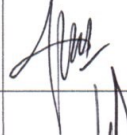
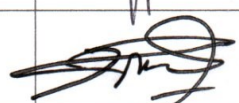
JL. Achmad Nadjamuddin No. 17 Telp. (0435) 829975 Fax (0435) 829976 Gorontalo

Berita Acara Perbaikan/Revisi Ujian SKRIPSI

Pada hari ini, Rabu 09-Desember-2020, Pukul 13.00-15.00 Wita. Telah dilaksanakan Ujian SKRIPSI mahasiswa Program Studi Teknik Informatika Fakultas Ilmu Komputer Universitas Ichsan Gorontalo.

Nama : Rizki Malahedi
Nim : T3116154
Pembimbing I : Irvan Abraham Salihi, M.Kom
Pembimbing II : Irma Surya Kumala Idris, M.Kom
Judul SKRIPSI : Game Simulasi Pembelajaran Budidaya Tanaman Jagung Berbasis Real Time Strategi

Oleh Komite Seminar sebagai berikut :

No	Komite Seminar	Status	Tanda Tangan
1	Budy Santoso, S.Kom, M.Eng	Ketua	
2	Abd. Rahmat Karim Haba, M.Kom	Anggota	
3	Warid Yunus, M.Kom	Anggota	
4	Irvan Abraham Salihi, M.Kom	Anggota	
5	Irma Surya Kumala Idris, M.Kom	Anggota	



KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN
UNIVERSITAS ICHSAN GORONTALO
FAKULTAS ILMU KOMPUTER

SK MENDIKNAS NOMOR 84/D/O/2001

Jl. Achmad Nadjamuddin No. 17 Telp. (0435) 829975 Fax (0435) 829976 Gorontalo

SARAN-SARAN PERBAIKAN

Berdasarkan SKRIPSI Mahasiswa

Nama : Rizki Malahedi

NIM : T3116154

Program Studi : Teknik Informatika

Yang diselenggarakan pada

Hari/Tanggal : Rabu / 09-Desember-2020

Waktu : 13.00-15.00 Wita

Tempat : R Ujian Lt2

Maka dapat disimpulkan saran-saran sebagai berikut:

-	Tambahkan uji Portability uji
	Varabel Uji : Portability.
	Output Uji : Layak, Sangat Layak, Tidak layak.

Demikian saran perbaikan SKRIPSI ini dibuat dengan sebenarnya untuk dapat dipergunakan sebagaimana mestinya.

Gorontalo, 09-Desember-2020

Dosen Penguji

Budy Santoso, S.Kom, M.Eng



KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN
UNIVERSITAS ICHSAN GORONTALO
FAKULTAS ILMU KOMPUTER

SK MENDIKNAS NOMOR 84/D/O/2001

JL. Achmad Nadjamuddin No. 17 Telp. (0435) 829975 Fax (0435) 829976 Gorontalo

SARAN-SARAN PERBAIKAN

Berdasarkan SKRIPSI Mahasiswa

Nama : Rizki Malahedi

NIM : T3116154

Program Studi : Teknik Informatika

Yang diselenggarakan pada

Hari/Tanggal : Rabu / 09-Desember-2020

Waktu : 13.00-15.00 Wita

Tempat : R Ujian Lt2

Maka dapat disimpulkan saran-saran sebagai berikut:

1	Sebaiknya flowchar dengan flowgraph
2	pengujian white box / Basispath dan flowgraph
3	Saran Bagi peneliti selanjutnya

Demikian saran perbaikan SKRIPSI ini dibuat dengan sebenarnya untuk dapat dipergunakan sebagaimana mestinya.

Gorontalo, 09-Desember-2020

Dosen Penguji

Abd. Rahmat Karim Haba, M.Kom



KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN
UNIVERSITAS ICHSAN GORONTALO
FAKULTAS ILMU KOMPUTER
SK MENDIKNAS NOMOR 84/D/O/2001

JL. Achmad Nadjamuddin No. 17 Telp. (0435) 829975 Fax (0435) 829976 Gorontalo

SARAN-SARAN PERBAIKAN

Berdasarkan SKRIPSI Mahasiswa

Nama : Rizki Malahedi

NIM : T3116154

Program Studi : Teknik Informatika

Yang diselenggarakan pada

Hari/Tanggal : Rabu / 09-Desember-2020

Waktu : 13.00-15.00 Wita

Tempat : R Ujian Lt2

Maka dapat disimpulkan saran-saran sebagai berikut:

-	Sesuai dengan Story board yang ada program.

Demikian saran perbaikan SKRIPSI ini dibuat dengan sebenarnya untuk dapat dipergunakan sebagaimana mestinya.

Gorontalo, 09-Desember-2020

Dosen Penguji


Warid Yunus, M.Kom



**KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN
UNIVERSITAS ICHSAN GORONTALO
FAKULTAS ILMU KOMPUTER**

SK MENDIKNAS NOMOR 84/D/O/2001

JL. Achmad Nadjamuddin No. 17 Telp. (0435) 829975 Fax (0435) 829976 Gorontalo

SARAN-SARAN PERBAIKAN

Berdasarkan SKRIPSI Mahasiswa

Nama : Rizki Malahedi

NIM : T3116154

Program Studi : Teknik Informatika

Yang diselenggarakan pada

Hari/Tanggal : Rabu / 09-Desember-2020

Waktu : 13.00-15.00 Wita

Tempat : R Ujian Lt2

Maka dapat disimpulkan saran-saran sebagai berikut:

	- tambahkan pengelompokan tugas & soal
	perbaikan.

Demikian saran perbaikan SKRIPSI ini dibuat dengan sebenarnya untuk dapat dipergunakan sebagaimana mestinya.

Gorontalo, 09-Desember-2020

Dosen Penguji

Irvan Abraham Salihi, M.Kom



KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN
UNIVERSITAS ICHSAN GORONTALO
FAKULTAS ILMU KOMPUTER

SK MENDIKNAS NOMOR 84/D/O/2001

JL. Achmad Nadjamuddin No. 17 Telp. (0435) 829975 Fax (0435) 829976 Gorontalo

SARAN-SARAN PERBAIKAN

Berdasarkan SKRIPSI Mahasiswa

Nama : Rizki Malahedi

NIM : T3116154

Program Studi : Teknik Informatika

Yang diselenggarakan pada

Hari/Tanggal : Rabu / 09-Desember-2020

Waktu : 13.00-15.00 Wita

Tempat : R Ujian Lt2

Maka dapat disimpulkan saran-saran sebagai berikut:

1.	Perbaiki pengujian sistem (white box)
2.	perbaiki segala revisi dan penguji

Demikian saran perbaikan SKRIPSI ini dibuat dengan sebenarnya untuk dapat dipergunakan sebagaimana mestinya.

Gorontalo, 09-Desember-2020

Dosen Penguji

Irma Surya Kumala Idris, M.Kom

Riwayat Hidup Peneliti



Nama : Rizki Malahedi

Tempat Tanggal Lahir : Paguyaman, 12 Oktober 1996

Pekerjaan : Mahasiswa

Email : rizkimalahedi@gmail.com

Riwayat Pendidikan:

1. TK Wonosari (2002)
2. SD Negeri 05 Wonosari (2003 - 2009)
3. MTs Al - Ihsan Patilanggio (2009 - 2011)
4. MTs Al - Maktab Wonosari (2011 - 2012)
5. MA Al Falah 1 Limboto Barat (2012 - 2015)
6. Universitas Ichsan Gorontalo (2016 - Sekarang)